CENTRO UNIVERSITÁRIO UNIVATES
CURSO DE SISTEMAS DE INFORMAÇÃO

PROJETO E DESENVOLVIMENTO DE UM SISTEMA PARA DEFINIÇÃO DE ASPECTOS E ANÁLISE DE SENTIMENTOS EM TEXTOS

Frederico Jacobi Sausen

Lajeado, novembro de 2015

Frederico Jacobi Sausen

PROJETO E DESENVOLVIMENTO DE UM SISTEMA PARA DEFINIÇÃO DE ASPECTOS E ANÁLISE DE SENTIMENTOS EM TEXTOS

Monografia apresentada no curso de Sistemas de Informação, do Centro Universitário UNIVATES, como parte da exigência para a obtenção do título de bacharel em Sistemas de Informação.
Área de concentração: Mineração de Opinião

ORIENTADOR: Evandro Franzen

Lajeado, novembro de 2015

Dedico este trabalho a minha esposa, meus pais e pessoas próximas, em especial pela dedicação e apoio em todos os momentos difíceis.

AGRADECIMENTOS

Agradeço primeiramente a Deus, por ter me escutados em todos os momentos difíceis, de ansiedade, medo, desistência, por ter iluminado os caminhos e aberto portas para a conclusão deste trabalho.

Agradeço o meu orientador pelo apoio e atenção prestada nas minhas dúvidas e esclarecimentos, nos inúmeros contatos realizados.

Agradeço meus familiares, amigos e conhecidos pela fé, força e compreensão dadas nesse período.

RESUMO

Opiniões possuem grande relevância no comportamento humano. As tomadas de decisões em organizações baseiam-se, muitas vezes, na opinião de seus *steakholders*. Já são inúmeras empresas que atuam na obtenção de opiniões a partir de plataformas como páginas web, redes sociais, blogs, fóruns e sites de avaliação de produto. Nestes ambientes é que a mineração de opiniões ou analise de sentimento pode ser aplicada com o objetivo de identificar automaticamente a opinião nas expressões, com flexibilidade e rapidez, realizando avaliações, polarizando sentimentos em uma grande massa de conteúdo na forma de texto. No processo de *data mining* ocorre a limpeza de ruídos, filtragem de informações, segmentação, atribuição de valores, lógicas de agrupamento, utilizando algoritmos de classificação, agrupamento, associação conforme objetivos e resultados. A busca pela classificação da polaridade no texto é baseada em diversos métodos de acordo com a necessidade ou cenário: O método léxico de classificação tem o pré-processamento simples nos textos com menos *tokens* e normalizações sem a necessidade de treinos, o método aprendizado de máquinas de processo supervisionado é sensível às atribuições das *features*, tem seu desempenho de acordo com a qualidade de dados treino. Este trabalho propõe o uso de técnicas de mineração de opiniões, a partir dos dados extraídos em sites de lojas eletrônicas, permitindo a definição de aspectos e polaridade aos dados, possibilitar a extração dos dados para arquivos de extensão arff, suportado pela ferramenta Weka. Execução de processo não-supervisionado de filtragem em conjuntos de dados para treinamento, permitindo a seleção, o pré-processamento e aplicação de algoritmos para definição do sentimento, gerando estatísticas para posterior análise.

Palavras-chave: Mineração de Opiniões, Classificação da Polaridade, Aprendizado de Maquinas e Sistemas de Informação.

ABSTRACT

Opinions have great relevance in human behavior. Decision making in organizations are based often on the opinion of steakholders. There are now numerous companies operating in obtaining opinions from platforms such as web pages, social networks, blogs, forums, product review sites. In these environments is that mining of opinions or feelings of analysis can be applied in order to automatically identifying the view in the expressions with flexibility and speed, conducting assessments, polarizing feelings in a large body of content in text form. In the data mining process occurs noise cleaning, information filtering, segmentation, assignment of values, logical grouping, using ranking algorithms, clustering, association as objectives and results. The search for the text polarity classification is based on various methods according to necessity or setting: The lexicon method of classification which has the simple preprocessing in the texts under tokens and normalization without the need for training, supervised process of machine learning method is sensitive assignments of features, has its performance according to the quality of training data. This paper proposes the use of opinions mining techniques, based on data extracted in electronic shopping sites, allowing the definition of the data paints and polarity, allow for the extraction of data files arff extension, supported by the Weka tool. Execution of unsupervised process of filtering data sets for training, allow the selection, preprocessing of data and the application of algorithms to define the feeling, generating statistics for later analysis.

Keywords: Mining opinions, Polarity Classification, Learning Machines and Information Systems.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

AM	Aprendizado de Máquinas
AMD	Análise em Mineração de Dados
AS	Análise de Sentimento
ARFF	Extensão arquivo WEKA
CRISP-DM	*CRoss Industry Standard Process for Data Mining* – Padrão de processo industriais cruzados para mineração de dados
DC	Descoberta de Conhecimento
EMIM	Expected Mutual Information Measure – Medida esperada para informação mútua
HTML	*HyperText Markup Language* - Linguagem de Marcação de Hipertexto
IA	Inteligência Artificial
JSON	*JavaScript Object Notation*
JVM	*Java Virtual Machine*
KDD	*Knowledge Discovery in Databases* - Descoberta de Conhecimento nas Bases de Dados
LSI	Indexação Semântica Latente
MD	Mineração de Dados
ME	*Maximum Entropy* – Entropia máxima
NB	*naive Bayes*
PLN	Processamento de Linguagem Natural
POS	Part of Speech
SGBD	Sistema de Gerenciamento de Banco de Dados
SVM	*Support Vector Machine* - Máquinas de Vetor de Suporte
SMO	*Sequential Minimal Optimization* - Mínima Optimização Sequêncial

TCC	Trabalho Conclusão de Curso
TI	Tecnologia da Informação
WEKA	Waikato Environment for Knowledge Analysis – Waikato ambiente de análise de dados de conhecimento

SUMÁRIO

1 INTRODUÇÃO

Com o crescimento das mídias sociais, *blogs* e *web*, a relação entre usuários está cada vez mais próxima e rápida. Devido a esta proximidade a opinião de alguém pode disseminar de forma rápida, tanto as boas opiniões quanto as más opiniões, podendo influenciar a organização nos processos de tomada de decisão. O gasto das organizações em *marketing* nas redes sociais, *web*, e-mail e outros, é cuidadosamente avaliado, pois a avaliação de apenas um usuário pode se tornar catastrófico para a reputação do produto ou empresa.

O grande volume de dados que circulam na web e estão armazenados em *Data Warehouses* e outros, é algo valioso para muitas empresas, tanto para o marketing, análise de dados, tomadas de decisão entre outros. A Mineração de Dados (*data mining*) tem papel fundamental na interpretação das informações e pode ser definida como a aplicação de técnicas para filtragem de informações, classificação ou segmentação de dados, entre outras operações.

O objetivo principal da Mineração de Dados (MD) é o de como transformar dados armazenados em conhecimento, expresso em termos de formalismos de representação, tal como regras e relações entre dados. Uma das formas de tornar os dados armazenados em conhecimento é utilizando algoritmos, muitos destes já disponíveis na web, através de diversas formas, existindo processos, etapas de testes e treinamento até chegarmos ao resultado final (REZENDE et al., 2003).

A partir da necessidade de extrair a opinião dos usuários de forma rápida, foram criados programas coletores de dados da web (*Web crawlers*) que extraem automaticamente textos da *web*, conseguindo visitar diversos sites coletando dados, transformados em informação e conhecimento, para posterior armazenagem e análise.

Mineração de Opiniões ou Analise de Sentimentos tem o objetivo de identificar, classificar e sumarizar sentimentos, opiniões, avaliações e resultados (SANTOS; BECKER; MOREIRA, 2014).

Uma opinião é definida por dois elementos em um documento: alvo da opinião e o sentimento expresso sobre o alvo, um documento sendo qualquer pedaço de texto em linguagem natural; o alvo também é conhecido como entidade, tópico ou aspecto sobre o qual foi manifestada a opinião de um produto, serviço entre outros, sendo polarizado entre bom e ruim (CARVALHO, 2014).

A classificação da polaridade do texto pode ser feita através de uma estratégia baseada em Aprendizagem de Máquina, que possui algoritmos para melhor desempenho de métricas baseado em treinamento de atributos quantitativos com aprendizado supervisionado ou não-supervisionado.

Este trabalho documentará o desenvolvimento de um sistema, denominado AMD (Análise em Mineração de Dados), para identificar aspectos em textos e para aplicar algoritmos de aprendizado de máquina. Serão utilizadas técnicas de mineração de opiniões, onde os dados serão dispostos para treino em um software elaborado pelo autor, onde serão atribuídas as polaridades dos textos manualmente criando um modelo de treino, modelo que será importado na ferramenta de mineração de dados Weka[1] como base para atribuição das polaridades.

1.1 Objetivos Geral

O objetivo geral deste trabalho é disponibilizar uma ferramenta que defina aspectos e aplique algoritmos para classificação de sentimento em textos.

1.2 Objetivos Específicos

- Desenvolver e testar uma ferramenta que permita a definição de aspectos, por exemplo: marca, modelo e tipo de produto; e classes, por exemplo: Positivo e Negativo; de forma prática ao texto manualmente.

[1] O Weka é um software bastante popular, sem custo, que possui uma coleção de algoritmos para aprendizado de máquina integrada, desenvolvida pela Universidade de Waikato (WAIKATO, 2015). Algoritmos esses que serão integrados à ferramenta desenvolvida neste trabalho, a ferramenta Weka é desenvolvida em Java e publicada sob licença GPL, assim as bibliotecas Weka podem ser chamadas por qualquer programa, facilitando a integração.

- Utilizar de recursos de outra ferramenta de mineração de dados para o pré-processamento, extração e pós-processamento dos dados.

- Exportar os dados para integrar os recursos desta ferramenta a uma ferramenta de mineração de dados escolhida, podendo assim ter flexibilidade na obtenção de resultados e estatísticas.

- Possibilitar a visualização dos aspectos e classes do conjunto de dados analisados, através de avaliações e resultados das classificações.

- Apresentar os principais conceitos relacionados aos recursos e módulos da ferramenta.

1.3 Justificativa

A mineração de dados é uma área que está em pleno crescimento, novos estudos, novos campos de atuação, tendência de mercado. A opinião se torna valiosa em determinadas situações para as pessoas ou para empresas terem conhecimento do que as pessoas estão achando de seu serviço ou produto.

Decidir entre qual algoritmo utilizar na mineração de dados é uma tarefa importante e que requer testes para identificar qual técnica se adapta melhor ao problema que terá de resolver. Para o uso correto dos algoritmos classificadores, é necessário ter um conjunto de dados treino rotulados, é um processo manual e exaustivo ter que atribuir aspectos e classes ao volume de dados suficientes, para que se tenha algum resultado significativo.

Construir uma ferramenta mais amigável ao usuário para classificação do conjunto de dados extraídos da *web* facilita o processo num todo da mineração de dados. A ferramenta desenvolvida permite a definição de estruturas variáveis para cada cenário a ser pesquisado, tornando inteligente a ferramenta para modelos de treinos, podendo não só pesquisar produtos eletrônicos, mas também opiniões de assuntos da atualidade, por exemplo: copa, eleições entre outros.

Como a mineração de opiniões é uma área recente, que pode ser pesquisada e analisada de diversas formas, flexibilizar a estrutura de pesquisa da ferramenta é importante, podendo ser utilizada para pesquisa de futuros trabalhos desenvolvidos no curso. Tornar o estudo da mineração de opiniões mais acessível e prático atrai mais adeptos à pesquisa.

Existem diversas ferramentas que abordam maneiras de decifrar a linguagem formal, para a língua portuguesa temos poucos dicionários (léxicos) disponíveis para mineração de opinião, as vezes se torna cansativo para o usuário manipular os dados, nesse pensamento o trabalho quer tornar essa prática amigável e acessível.

Gerar estatísticas pode ser importante para que empresas tenham apoio na tomada de decisão, evitando escolhas equivocadas de técnicas de mineração de dados para solução de problemas, sem a necessidade de técnicas de mineração de dados complexos, ferramentas inadequadas, evitando altos custos desnecessários. Muitos sistemas existentes possuem alto custo de contratação e manutenção de seus serviços específicos para a área.

1.4 Estrutura do documento

Este trabalho está estruturado da seguinte forma:

Capítulo 2 - Descoberta de Conhecimento: será abordado todo o referencial teórico da proposta do trabalho, *data mining*, mineração de opiniões e uma documentação de ferramentas de mineração de dados.

Capitulo 3 - Trabalhos relacionados: será apresentado trabalhos que de alguma forma contribuirão para a proposta deste trabalho.

Capitulo 4 – Materiais e Métodos: será abordada a metodologia utilizada e tecnologias utilizadas.

Capitulo 5 – Resultados e Discussão: o capítulo mostrará as telas desenvolvidas na ferramenta, testes com as ferramentas e situações percebidas.

Capitulo 6 – Conclusão: será discutido os objetivos do trabalho, o que foi desenvolvido na ferramenta e trabalhos futuros.

2 DESCOBERTA DE CONHECIMENTO

Segundo Dias (2002), desde a invenção dos computadores, o principal objetivo de utilizar os computadores é trazer solução aos problemas operacionais das empresas, mas ainda existem empresa sem meios de utilização de computadores nas tomadas de decisões. Ainda existem dificuldades na descoberta de conhecimento baseada na grande quantidade de informações em banco de dados que as empresas possuem, relacionadas a fatores como: na mineração de dados há falta de conhecimento da existência de técnicas, técnicas complexas, falta de ferramentas adequadas, alto custo das ferramentas de mineração, escolha errada da técnica conforme problema a ser resolvido. As técnicas de mineração são aplicadas em sistemas de descoberta de conhecimento em banco de dados, na intensão de obter informações estratégicas, por meio de pesquisas dessas informações, padrões e de classificação e associação entre outros.

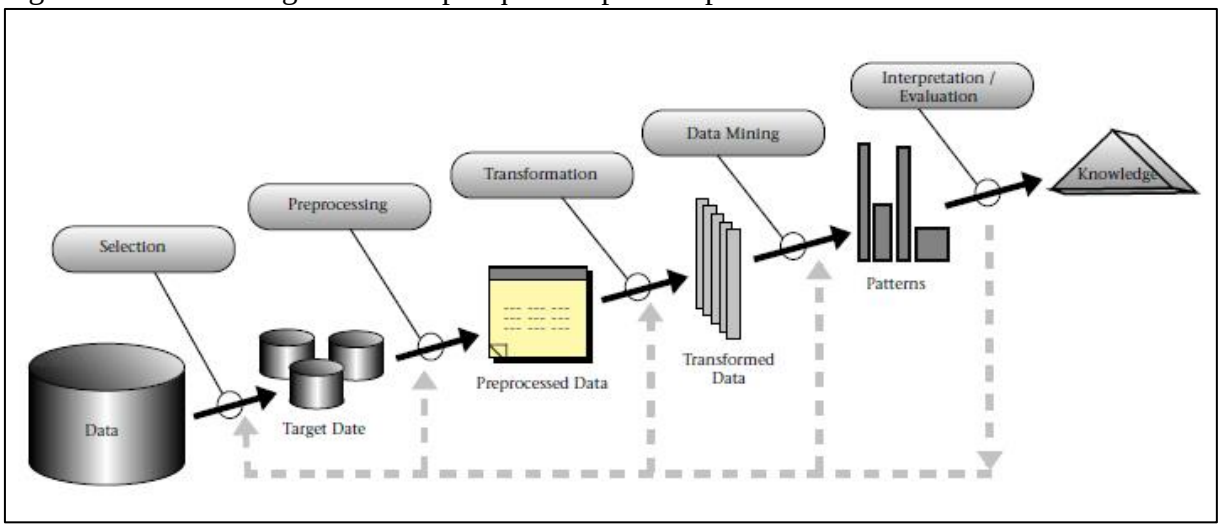
O processo *Knowledge Discovery in Databases* (KDD), a Mineração de Dados está em uma das suas fases, na qual ocorre a aplicação de algoritmos que identificam padrões válidos, novos, potencialmente úteis e compreensíveis. O processo se preocupa com o desenvolvimento de métodos e técnicas para fazer sentido aos dados. (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996).

O método tradicional de se transformar os dados em conhecimento baseia-se na análise manual e interpretativa, porém a descoberta de conhecimento refere-se ao processo global de descoberta de conhecimento a partir de dados e mineração de dados refere-se a uma etapa neste processo, a mineração de dados é uma aplicação de algoritmos específicos para extração de padrões de dados, existe uma distinção entre o processo KDD e a etapa de Mineração de Dados. (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996).

Segundo Camilo (2009), existem diversos conceitos relacionados à KDD e *Data Mining*. Alguns autores consideram eles sinônimos e outros autores definem o KDD com um processo da descoberta de conhecimento e o *Data Mining* se refere a atividade do processo. Somente concordando que o *Data Mining* precisa ser divido em fases, iterativo e interativo.

O processo de KDD é interativo e iterativo, envolve diversas etapas e diversas decisões feitas pelo usuário, para se ter uma visão prática do processo, destaca-se as etapas na Figura 1.

Figura 1 - Uma visão geral das etapas que compõem o processo de KDD.



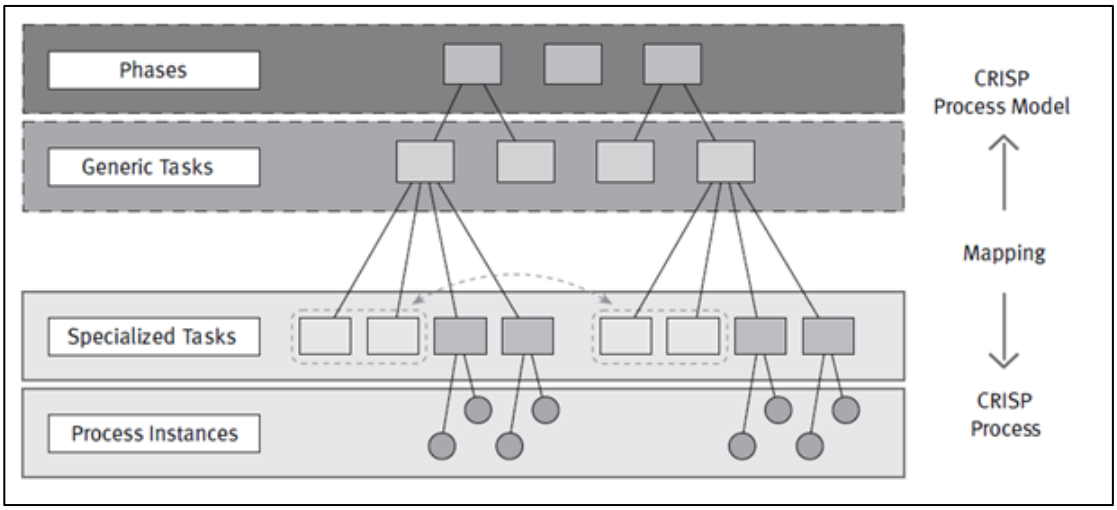
Fonte: FAYYAD; PIATETSKY-SHAPIRO; SMYTH (1996).

A Mineração de Dados é constituída por diversos processos com fases e atividades, contendo a mesma estrutura. Com o intuído de padronizar as etapas do processo, surge o projeto CRISP-DM (*CRoss Industry Standard Process for Data Mining*), que auxilia no desenvolvimento da proposta do trabalho.

2.1 Processo CRISP-DM

A metodologia CRISP-DM é descrita como um modelo de processo hierárquico, composto por um conjunto de tarefas, em quatro nível de abstração: Primeira fase, em seguida Tarefas Genéricas, Tarefas Específicas e Instância de Processos como mostra a Figura 2 (CHAPMAN, 2000).

Figura 2 - Quatro níveis de desagregação da metodologia CRISP-DM



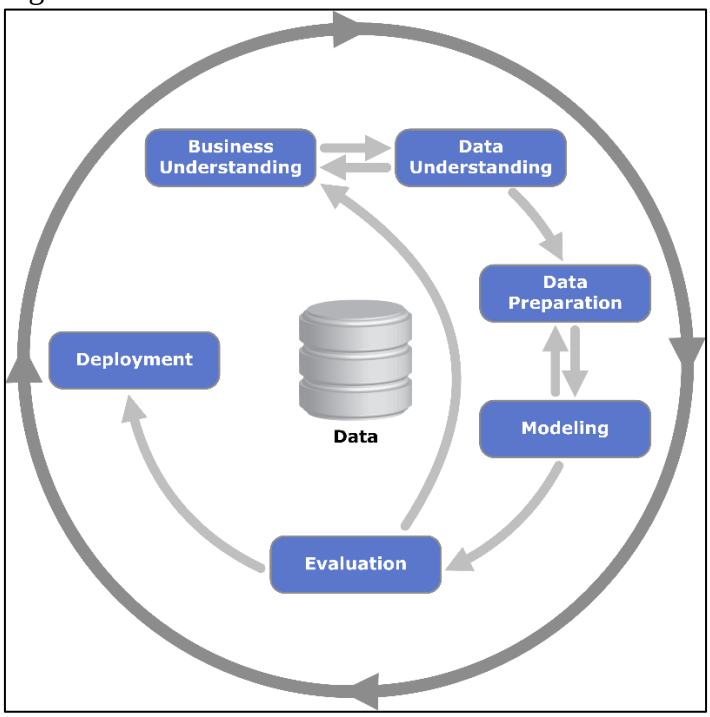
Fonte: CHAPMAN (2000).

Na primeira fase os processos de mineração de dados são organizados em uma série de fases, um conjunto de fases consiste nas tarefas genéricas, onde cobre todos os dados possíveis de situação para mineração. No nível das tarefas especializadas é definido como as ações do segundo nível serão aplicadas, por exemplo: no segundo nível, possui uma tarefa genérica limpa dados, o terceiro nível descreve essa tarefa em diferentes situações, como valores numéricos agiriam com valores categóricos. O quarto nível, instância de processos, são registradas as ações, decisões e resultado de uma mineração de dados, é organizado de acordo com as tarefas definidas nos níveis mais altos, representando o real acontecimento do conjunto.

O CRISP-DM foi desenvolvido baseado nos primeiros processos de descoberta de conhecimento, utilizando os requisitos dos usuários, sendo adaptável a diversos setores da indústria. Tornando pequenos e grandes projetos mais rápidos, baratos, confiáveis e gerenciáveis. Sendo referência no desenvolvimento de um plano de integração na DC (AMORIM, 2006).

O modelo atual do CRISP-DM no processo de Mineração de Dados apresenta uma visão geral do ciclo de vida, destacando as relações entre as tarefas, podendo existir diversas tarefas de acordo com o os objetivos, dados e outros. O projeto de mineração de dados é dividido em 6 fases o ciclo de vida, como mostra a Figura 3.

Figura 3 - Fases do modelo de referência CRISP-DM



Fonte: CHAPMAN (2000).

A sequência das fases não é obrigatória, em determinado momento pode haver diferença na sequência de fases. O resultado de cada fase ou uma tarefa em particular da fase, determinada a próxima fase. As setas indicam a sequência padrão entre as fases. O ciclo externo indica a sequência natural da mineração de dados em si. Um ciclo de mineração de dados pode iniciar outro ciclo de processos, levando conhecimento e soluções, beneficiando processos posteriores (CHAPMAN, 2000).

2.1.1 Entendimento do negócio (*Business Understandin*)

Fase onde são levantados os objetivos e requisitos do projeto, a partir do conhecimento do negócio que irá ajudar nas próximas etapas, através destas informações definir o problema de mineração de dados e traçar um plano para atingir os objetivos.

2.1.2 Seleção dos dados (*Data Understanding*)

Nesta etapa são realizados os agrupamentos de dados, verificado qualidade de dados, descobrimento dos primeiros *insights* sobre os dados, e detecção de subconjuntos para formação de hipóteses sobre informações ocultas.

2.1.3 Limpeza dos dados (*Data Preparetion*)

Fase onde já se tem um pacote de dados finais, dados que serão adicionados a ferramenta de mineração, a partir do pacote bruto de dados. Ocorre a limpeza, transformação e formatação dos dados conforme dados de etapas anteriores, nesta fase os ruídos e dados estranhos são moldados e tratados.

2.1.4 Modelagem dos dados (*Modeling*)

Nesta fase as técnicas de mineração são selecionadas e aplicadas nos valores, seus parâmetros são calibrados para obter valores ideais. Podem existir diversas técnicas para o mesmo problema de mineração de dados. É preciso as vezes voltar para a etapa anterior, algumas técnicas possuem requisitos na forma de dados.

Segundo Camilo (2009) muitas técnicas de mineração de dados são conceituadas de aprendizagem de máquina, reconhecimento de padrões, estatísticas, classificação e clusterização.

2.1.5 Avaliação do processo (*Evaluation*)

Na avaliação do processo acontece uma fase crítica, na qual os especialistas no negócio e tomadores de decisões são necessários. Ferramentas gráficas são utilizadas para visualização e análise de resultados, sendo avaliada a possibilidade de retornar a fases anteriores do processo, sempre verificando se o modelo de dados atingirá os objetivos do negócio. O mais importante desta fase é verificar se algum dos problemas de negócio ainda não foi detectado. No final, tendo a decisão da utilização dos dados nos resultados.

2.1.6 Execução (Deployment)

Esta fase não é geralmente o fim do projeto. O conhecimento adquirido até então, deverá ser apresentado para o cliente na forma que ele entenda a informação. Pode ser tanto na forma de páginas da Web ou execução dos processos de mineração de dados iguais em diversos setores da organização. Passar ao cliente todas as informações adquiridas para que utilize corretamente os modelos criados.

2.2 Data Mining

Data Mining, ou Mineração de Dados, é a ação em um grande conjunto de dados, em busca de informações, exploração e conhecimento de dados, estabelecendo regras de associação, limpeza de ruídos, atribuição de valores, lógicas de agrupamentos, onde são utilizados algoritmos de acordo com o objetivo final.

No processo de mineração a coleta, a aplicação dos algoritmos e visualização dos dados são divididos em diversas etapas, não sendo um sistema automático. Etapas onde é feito o conhecimento do domínio do que será minerado, o pré-processamento, a extração de padrões, pós-processamento e a utilização do conhecimento.

Usuários do processo são divididos em três classes: especialista do domínio, o que possui conhecimento do domínio, dando apoio na execução do processo; analista, usuário especialista no processo de Extração de Conhecimento, responsável pela sua execução, com conhecimento profundo das etapas e; usuário final, que utiliza o conhecimento extraído, auxiliando-o no processo de tomada de decisão (REZENDE et al., 2003).

2.2.1 Conhecimento do Domínio

Conhecimento no conteúdo, domínio abordado, definição das metas e objetivos serão traçados nesta etapa, tendo papel importante no fornecimento de conhecimento do conteúdo, dando apoio ao analista com a extração de padrões. Fornecendo assim um suporte nas próximas etapas do processo, como o pré-processamento, ajudando o analista no conjunto de dados, valores válidos, preferências, restrições para os atributos.

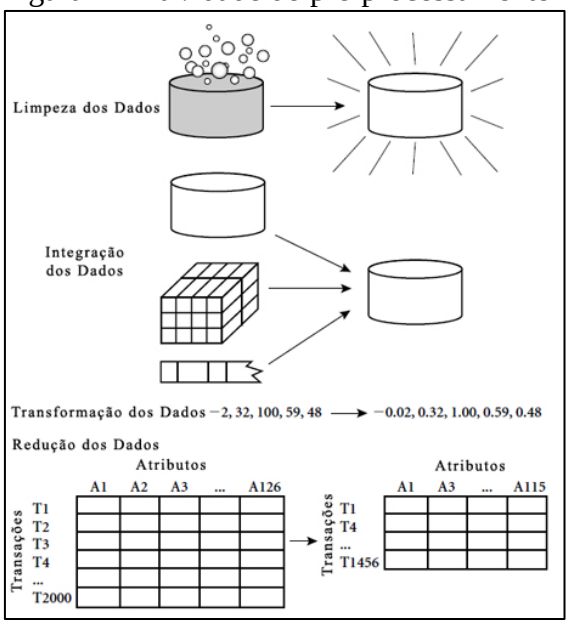
2.2.2 Pré-processamento

Antes de aplicar a mineração de dados, precisamos saber com que tipo de dados esta trabalhando, preparar esses dados se torna importante, através do especialista definir pesos aos dados pré-visualizados, definindo as técnicas a serem usadas. Além de definir as informações corretas, as definições das informações incorretas também precisam de atenção como, os nulos, em branco e *emotions* devem ser categorizados.

Conforme Figura 4 o pré-processamento consiste na limpeza dos dados, diversas vezes os dados são encontrados com inconsistências, ocorre então a avaliação de dados, evitando

futuros problemas nos algoritmos. Na limpeza os dados podem ser removidos, agrupados e até ganhar atributos (CAMILO, 2009).

Figura 4 - Atividade do pré-processamento



Fonte: CAMILO (2009).

A ferramenta Weka, que é documentada no capítulo 2.4.1, possui o algoritmo filtro não-supervisionado StringToWordVector, algoritmo que converte os atributos do tipo String em um conjunto de atributos que representam as ocorrências das palavras, informações baseadas nas frases contidas nos textos (WAIKATO, 2015). Este algoritmo possui em suas propriedades outros algoritmos de *tokens*, *stemmer* e *stopwords*.

Nos textos existem termos frequentes que não carregam nenhuma informação de maior relevância, são as *stopwords*, as quais são compostas por palavras das seguintes classes gramaticas: artigos, preposições, conjunções, pronomes e advérbios. A remoção das *stopwords* tem como objetivo eliminar termos não representativos ao texto, essa técnica também é considerada como compressão de textos, já que reduz as palavras analisadas no texto e o número de palavras a serem armazenadas no banco de dados (DIAS; DE GOMENSORO MALHEIROS, 2005).

Há depois a necessidade de integração com os dados, com a diversidade de possibilidades de coletas, como fotos, vídeos, bancos de dados, web sites, planilhas, surge a necessidade regras de associações para os mesmos registros vindos de locais diferentes, mesmos registros recebendo atribuições diferentes para no final termos um local único, com registros consistentes e categorizados.

Ocorre então a transformação dos dados, podendo os dados se tornar numéricos ou categóricos, depende do algoritmo utilizado. De acordo com o objetivo da mineração, a técnica e critérios são escolhidos. Técnicas como suavização, generalização são escolhidas na transformação de dados (REZENDE, 2005).

Nas situações de grande volume de dados, há a possibilidade de redução dos dados, sem comprometer a integridade do resultado original. Facilitando a vida dos algoritmos, tornando-os eficientes e com qualidade. Estratégias como cubo de dados, redução de dimensionalidade são utilizadas nessa etapa (CAMILO, 2009).

2.2.3 Extração de Padrões

Nesta etapa os objetivos traçados na identificação do problema tendem a ser cumpridos. Etapa de processo iterativo, pode ser necessário à sua execução diversas vezes até atingir o objetivo preestabelecidos, sendo possível haver ajustes para melhor precisão e conhecimento dos dados (REZENDE, 2005).

Há duas escolhas nos algoritmos de tarefas a serem utilizadas, preditiva e descritiva. Na preditiva consiste na diversidade de exemplo e minerações passadas com respostas semelhantes. Segundo Rezende (2005), na descritiva identifica-se comportamentos intrínsecos, dados com só uma classe especificada.

A escolha do algoritmo está baseada nos resultados a serem encontrados. Algoritmos mais simples são de melhor interpretação, já algoritmos complexos são utilizados por pesquisadores.

Com os dados o algoritmo de extração poder ser executado várias vezes, dependendo da função escolhida. Nos diversos métodos estão sendo feitos agrupamentos de informações e sofisticação de técnicas entre extrações, assim obtendo melhores resultados e menores índices de erro.

2.2.4 Pós-Processamento

A busca por conhecimento no banco de dados é exaustiva, o resultado obtido pelo algoritmo pode as vezes conter inúmeros padrões, com redundâncias, muitos destes sem valor algum ao usuário. O usuário quer encontrar uma pequena listagem, algo que ele consiga

entender e seja objetivo, então disponibilizar um grande número de padrões pode ser decepcionante à sua expectativa.

Para contornar essa situação estão sendo pesquisadas medidas para entendimentos e conhecimentos adquiridos. Medida subjetiva tem sua função quando usuários possuem diferentes objetivos para um padrão, onde aqui são avaliados interesses específicos para elaboração de regras de cada usuário.

Com os resultados apresentados e o usuário estiver insatisfeito com os padrões divulgados, o processo de extração pode ser executado novamente, verificando dados, adaptado o processo, para que na próxima extração se tenha algo melhor a ser apresentado.

2.3 Mineração de Opiniões

A influência da opinião entre nós pessoas tem grande impacto, saber a opinião de outra pessoa para as atitudes ou dúvidas tem interferência nas tomadas de decisões. Na escolha de uma roupa, qual livro ler, o filme que está passando no cinema. Nas organizações, saber a opinião dos clientes é algo importante, tanto na satisfação como no lançamento de novos produtos ou serviços.

Existem organizações que trabalham só na obtenção destas informações. Como as opiniões de alvo público que são tradicionais, trabalho que é executado envolvendo diversos profissionais, com saída de campo, questionários, telefonemas entre outros. A respostas destas ferramentas são adquiridas em um tempo bastante longo e exaustivo, para então envolver outra demanda de profissionais, onde se tem uma demora no agrupamento e ordenação do que foi coletado, organizando os dados para serem apresentados os resultados do serviço.

Com o surgimento das redes sociais, a opinião dos usuários se tornou mais contínua e de fácil obtenção. Ter acesso a opinião de outras pessoas se tornou muito fácil, não se restringe mais ao grupo de convívio, o grupo de pessoas se torna grandioso, se obtém acesso a opiniões de profissionais da área, onde antes só de revistas e planilhas (BECKER, 2014).

O estudo de redes sociais está na sua infância, nas diversas áreas vem surgindo novos problemas, desafios e oportunidades de pesquisa. Na computação vem se tornando tema chave em pesquisas, incluindo sistemas multimídias, recuperação de informação, mineração

de dados e aprendizagem de máquina. As redes sociais permitem discussões em larga escala sobre questões de diferentes áreas (BENEVENUTO; ALMEIDA, 2011).

As redes sociais possuem estudos focados na identificação e monitoração da polaridade em *post* compartilhados, partindo da hipótese de que a quantidade expressiva de *post* está relacionada ao humor e a emoções expressas pelo usuário (ARAÚJO; GONÇALVES; BENEVENUTO, 2013).

Nesse cenário a mineração de opinião automática começa a criar forma. Começam a surgir mecanismo de obtenção e manipulação deste grande volume de informações, algoritmos que extraem os dados, agrupam as informações e disponibilizam de forma prática, em um tempo muito menor, com um número de conhecimento muito maior.

> A mineração de opiniões possui dois objetivos principais [47]: (i) identificar e discernir entre documentos que contêm fatos (notícias, por exemplo) e documentos que contêm opiniões, e (ii) classificar as opiniões quanto às suas polaridades, ou seja, se são opiniões positivas ou negativas (CARVALHO, 2014, p. 1).

Textos que expressam sentimentos postados em *blogs*, redes sociais, fóruns, *twetts* e outros são fontes ricas para Análise de Sentimento ou Mineração de Opinião, com o objetivo de identificar, classificar e sumarizar sentimentos do texto, referente a um alvo (SANTOS; BECKER; MOREIRA, 2014).

Os problemas que a mineração de opinião encontra hoje é decifrar as estruturas dos textos, encontrando conteúdos de acordo com o que se está procurando, classificar a polarização (definir entre bom ou ruim) das informações declaradas, agrupar essas informações, organizar no entendimento do usuário final (BECKER, 2014). Diversas técnicas estão sendo criadas para facilitar esse trabalho, como opiniões falsamente lançadas, definir pontos de divulgação de produtos, monitoramento de entidades, marketing de produtos, relacionamento com o cliente.

2.3.1 Análise de Sentimento

Análise de Sentimento (AS) identifica nos textos opinativos o sentimento expresso, por exemplo: textos com opiniões de objetos ou tópicos de interesse, textos opinativos são considerados subjetivos. As opiniões podem ser positivas, negativas ou neutras, indicando sua polaridade, onde nos textos são expressas por palavras opinativas, por exemplo: adjetivos =

ruim, bom; advérbios = mal, devagar; e substantivos = amigos (SILVA; LIMA; BARROS, 2012).

A opinião é formada por dois elementos: o *alvo* e *sentimento*, *alvo* pode ser representado por um aspecto como produto, pessoa, empresa, assunto entre outros, o *sentimento* seria a polaridade deste *alvo*, com classes descritivas como positivo, negativo e neutro ou classes numéricas (1, -1).

A maneira de como será processada a opinião está na sua forma de expressão. Opiniões podem ser regulares ou comparativas, diretas ou indiretas, e implícitas ou explicitas (BECKER, 2014), regular quando o autor expressa sua opinião sobre o alvo, por exemplo: Este notebook é ótimo, na comparativa é expressado a similaridade, diferença ou afeição entre duas ou mais entidades, por exemplo: O sinal *wifi* deste *smartphone* é melhor que do meu anterior, comparando o sinal com o antigo. Na opinião direta, o sentimento é perceptível ao alvo, por exemplo: Meu novo carro é maravilhoso, na indireta o sentimento tem alvo no efeito da entidade pesquisada, por exemplo: O carro está pior depois do concerto da mecânica. E opiniões explicitas expressam o sentimento do alvo, por exemplo: “Picolé delicioso”, na implícita expressa ironicamente o sentimento, por exemplo: “Minha roupa encolheu depois que lavei”.

Palavras que expressam diretamente o sentimento do usuário é algo comum, mas interpretar palavras que possuem duplo sentido é algo rotineiro, ou destacar a situação de um alvo sem palavras explicitas de sentimento, por exemplo: Meu criado mudo molhou e já inchou, ou expressões ditas ironicamente ou subjetivamente.

2.3.2 Processamento de Linguagem Natural

Segundo Becker (2014), os recursos do processamento de linguagem natural (PLN) na mineração de opiniões tem fundamento básico, trata computacionalmente diversas questões da comunicação humana, como formatos, estruturas, significados, contextos e usos, têm desafios relacionados à compreensão da linguagem, oral ou escrita.

Revolucionar a maneira que o computador é usado, é uma motivação para estudos relacionados ao PLN, computadores que conseguiram entender a linguagem natural, puderam utilizar essa informação de forma rápida. A linguagem é um aspecto fundamental no

comportamento humano, na forma escrita passa de geração para geração e na forma falada é um meio de comunicação entre pessoas (DA SILVA; MARTINS, 2008).

O PLN sofre grandes faltas nas soluções de alguns problemas na Análise de Sentimento, é um problema mais restrito em vista que não há necessidade de compreensão total de sentenças e texto, como linguagens informais, por exemplo: expressões da internet, e sim compreensão das opiniões e seus alvos (BECKER, 2014).

A tokenização é algo essencial para as tarefas do PLN e de grande importância para a mineração de opinião, já que os dados de sentimentos são poucos, onde muitas vezes os dados são expressos de maneira singular, por exemplo os *emotions*: =). Existem várias regras de segmentação de textos, cada algoritmo adota sua linha de raciocínio, há tokenizadores que segmentam baseado em espaços, tabs, nova linha, caracteres de pontuação, nomes compostos e outros.

Um processo importante no PLN são os etiquetadores de texto (*taggers*), utilizados na etiquetagem automática de *corpus*[2], base fundamental se falando em AS. Essas etiquetas, em inglês termo definido como *Part of Speech* (POS), são classes gramaticais (morfossintáticas) das palavras do *corpus*. É definido um número finito de etiquetas (*tags*) tendo um significado linguístico associado, o significado da etiqueta vem da categoria morfossintática ou gramatical da palavra. O etiquetador define os *tokens* do texto com suas respectivas classes, em uma única palavra pode haver diversas rotulações.

Esse processo de etiquetagem associa palavras que estão em um certo texto a uma etiqueta do conjunto finito de etiquetas, sendo realizado de acordo com o algoritmo do etiquetador (DA SILVA; MARTINS, 2008), sendo executado com três componentes: escrutinador léxico[3], onde identifica símbolos, por exemplo: pontuação, palavras da língua ou estrangeiras entre outros; o classificado gramatical que atribui classes gramaticais às palavras; e desambiguizador, resolvendo ambiguidades léxicas, palavras com mesma grafia, mas significados diferentes.

É importante ressaltar expressões de negação nos textos, palavras que mudam o sentido da opinião ou sentimento. Palavras como “bom” e “ruim” pode ter seu sentido

[2] (Plural: corpora) Conjunto de texto cuidadosamente coletados, para diversas áreas, contendo dados autênticos, originados do computador, sendo representados pelas línguas utilizadas no estudo (RODRIGUES, 2009).
[3] Dicionário de Polaridade.

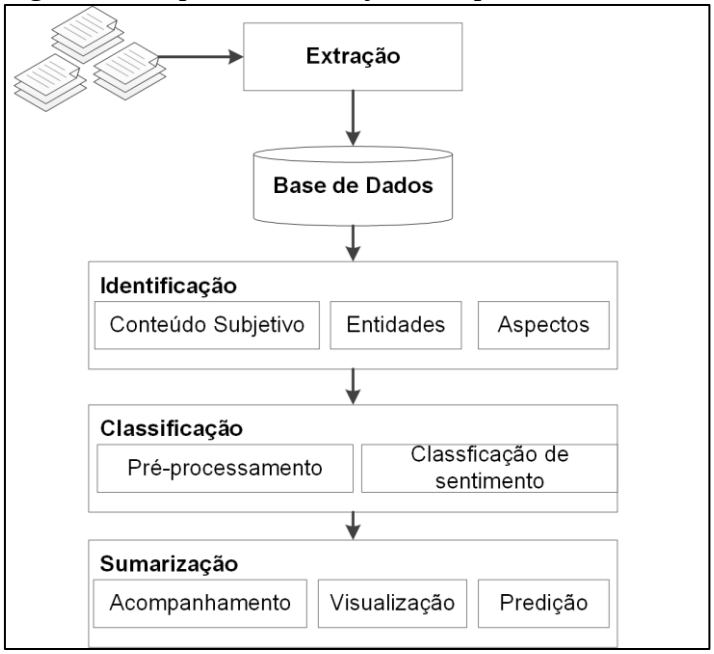
invertido, por exemplo: “A televisão não é boa” ou “O carro não é ruim”. Ironia também é algo bastante difícil de se tratar (BECKER, 2014).

A co-referência é algo que requer métodos bastantes atenciosos, uma palavra pode ter várias referências, por exemplo: Chefe, Patrão, Gerente, Seu José, todas essas palavras se referem a um sujeito. Destaque aqui para os pronomes que podem ter co-referência com outra palavra própria.

2.3.3 Etapas da Mineração da Opinião

Como mostra a Figura 5, a mineração de opinião tem como características 3 tarefas: Identificar, Classificar e Sumarizar (BECKER, 2014).

Figura 5 - Etapas da Mineração de Opinião



Fonte: BECKER (2014).

A tarefa de identificação está associada à coleta de dados dos textos, localizando os tópicos do conteúdo a ser analisado. As dificuldades e barreiras nessa tarefa estão na origem da coleta, por exemplo: página ou mídia, ou na forma que os dados estão dispostos. Aliás, as aplicações mais comuns são minerações de opiniões em produto ou serviços pela sua facilidade na busca de locais para extração de dados, a dificuldade está só em identificar os aspectos da entidade.

Quando não há uma entidade definida e contém diversas opções em um texto, pode ser definido entidades pré-definidas. Outro problema na identificação dos dados é co-referência,

já comentada na seção 2.3.2, expressões citadas que referenciam outras palavras, por exemplo: apelidos, gírias locais, *hashtags* e outros.

Na tarefa de classificação da polaridade ou sentimento, a tarefa possui uma classificação binária, com classes positivas ou negativas, em algumas consultas de dados pode se tornar problemas, caso o usuário queira algo mais detalhado, mas não há problemas em adicionar mais classes conforme necessidade. Na classe neutra, não são encontradas tendências de polaridades claras nos textos minerados.

Transformar os dados coletados e minerados em algo que o usuário entenda é a função da tarefa de sumarização, converter para métricas e sumários, com finalidade de tornar os dados em números de diversidades de opiniões da entidade, métricas que mostram o potencial do sentimento nos dados, por exemplo: produtos recém lançados no mercado, onde podem ser visualizados pelos usuários ou servir como conteúdo para outras aplicações.

2.3.4 Polaridade Baseada em Dicionário

O termo dicionário remete a objetos, por exemplo: impressos ou eletrônicos; para leitura humana e disponíveis em máquinas. No entanto para se ter informação e conseguir processá-la, ela cada vez mais é disponibilizada por léxicos, que na linguagem computacional da língua, refere-se a um componente de um sistema com informações (semântica e/ou gramatical) sobre palavras ou expressões (FREITAS, 2013), podendo partir de palavras sementes, como grupos menores de palavras com polaridade específica, por exemplo: bom, ótimo;

A composição básica de um léxico de sentimento é: palavra de sentimento e sua polaridade, expresso em categoria ou escala, relacionados a um idioma específico (BECKER, 2014). Existem bases lexicais já existentes, como para língua inglesa General Inquirer, OpinionFinder, SindiWordNet, para o português temos, OpLexicon e SentiLex-PT, existem alguns léxicos com múltiplas línguas.

Um método também utilizado é co-ocorrência entre palavras, possibilidade de palavras similares tendem a co-ocorrerem. Definindo palavras sementes com co-ocorrência no mesmo documento, pode ser utilizada uma ligação entre elas, para as classes positivas e negativas correspondentes. Por exemplo: “o notebook é ótimo”; a palavra “ótimo” possui uma polaridade positiva vinculada ao notebook, assim a co-ocorrência apresenta um resultado

satisfatório a nível textual quando pequeno, já que a entidade está próxima da entidade que qualifica.

2.3.5 Polaridade Baseada em Aprendizado de Máquina

O Aprendizado de Máquina (AM) possui algoritmos com objetivo de melhorar o desempenho da capacidade do modelo de prever corretamente (métrica acurácia de classificação) a partir da experiência (ARRIAL, 2008), esses algoritmos precisam de treinamentos prévios através de um conjunto de treinos de atributos quantitativos, representando informações ou características dos dados, baseados no aprendizado supervisionado ou não-supervisionado, esses dados são utilizados pelo algoritmo para selecionar regras que irão identificar ou classificar os comentários posteriores.

Comentários com opiniões prós e contras, na maioria das vezes são utilizados algoritmos de AM, que tenham *corpus* de exemplos etiquetados (SILVA; LIMA; BARROS, 2012). Cita-se alguns algoritmos como *naive Bayes* (NB), baseado na aplicação do Teorema de Bayes com a ideia de independência entre variáveis. A probabilidade de ocorrer cada classe depende dos valores de cada *feature*, por exemplo: a classe “cardíaco”, temos as *features* “colesterol” e “alimentação”, ambas não possuem relação uma com outra, mas dependendo de seus valores detectamos a classe “cardíaco” (BECKER, 2014).

O algoritmo *Support Vector Machines* (SVM) chama atenção pelos resultados: possui muita assertividade, tem interpretação simples de situações não-lineares complexas, sendo utilizado em relações lineares e não-lineares, entre outros, tem aceitação em tarefas de classificação e predição, existem muitas pesquisas no tempo de aprendizado que é um dos problemas desta técnica (CAMILO, 2009).

O C4.5, algoritmo que segue princípios de árvore de decisão, tem o método conhecido como “dividir para conquistar” uma pesquisa top-down mostrando possíveis caminhos na árvore, comparando com o exemplo disponibilizado. O critério de decisão de qual atributo será adicionado ao nó de decisão é feita através do ganho de informação (*information gain*), o ganho tem como medida a avaliação de quanto um atributo pode separar um conjunto de exemplos em categorias, o atributo que tiver o maior ganho será adicionado a árvore (SCHMITT, 2013).

Maximum Entropy (ME), outro algoritmo classificador probabilístico, ao contrário de NB, ele não assume a independência das *features*, em relação aos textos torna-se interessante, pois nem todas as palavras são independentes uma das outras. No ME dados de treino são usados como um conjunto de restrições sobre distribuição uniforme em todas as classes, exceto em casos específicos de dados de treino. O ME tem a desvantagem de não ter o mesmo desempenho quanto o NB, pelo motivo da otimização necessária para estimar os parâmetros do modelo, mas mostra eficiência nos problemas não binários, por exemplo: com mais de duas classes. (BECKER, 2014).

O algoritmo NBTree, o qual induz um modelo de classificação híbrida, uma combinação de árvores de decisão com *naive Bayes*, árvores com nodos classificadores *naive Bayes*, mantendo a boa interpretabilidade das duas técnicas. Induz classificadores altamente precisos e supera significamente as técnicas que os constituem (C4.5 e *naive Bayes*). O NBTree gerou classificadores com alta acurácia para dados reais e de larga escala (BASGALUPP, 2010)

Muitos dos resultados adquiridos pelos algoritmos possuem influência da preparação e seleção das *features*[4] na etapa de treinos (BECKER, 2014). Uma modalidade de preparação é com palavras de sentimento, alguns trabalhos limitam os termos com expressões de sentimento, assim os adjetivos e advérbios tem potencial no texto, algumas situações verbos e substantivos também, um léxico ajuda e muito na conclusão da contagem de termos de sentimento.

Outra preparação que vale destacar é a representação binária ou pesos, no lado da binária a ideia é informar se a expressão aparece ou não aparece, por exemplo: 0 ou 1; na frase, texto menores como *tweets*, post é bastante usado. No lado dos pesos a ideia é ao contrário, análise a nível de documento e não de textos menores, são registrados a frequência que a expressão aparece no texto, formalizada pelo tamanho ou no conjunto do documento.

Outra preparação é a normalização e POS tendo escolha pelo uso dos radicais ou dos lemas, o uso de radicais não tem uma boa aceitação por parte da polaridade, pois podem eliminar as variantes do mesmo radical, alterando a polaridade do sentimento original. Utilizar o POS para diferenciar entre expressões semelhantes é bem vindo, por exemplo: corr_V e corr_ADJ, pelas expressões “correr” e “corredor”.

[4] Atributos dos dados relevantes para definição dos vectores de suporte.

2.3.6 Redes Sociais Digitais

De modo geral, as redes sociais digitais se tornaram uma febre mundial se falando em informação e comunicação entre pessoas, o acumulo de informações publicadas contendo opiniões é gigantesco. Com a internet os usuários se sente mais seguros em declarar suas opiniões referente à algum acontecimento ou produto, onde cara a cara poderia haver uma certa intimidade (SANTOS, 2011).

Nas redes sociais é que se encontram os maiores gargalos quando o assunto é interpretação gramatical ou vocabulário, devido as abreviações, gírias, termos regionais, *emotions* entre outros. São textos mais específicos e com opiniões inconstantes.

Twitter[5] e Facebook[6] estão entre as redes sociais digitais mais utilizadas, segundo (BRITO, 2015) o twitter divulgou o resultado do primeiro trimestre de 2015, a rede social atingiu 302 milhões de membros, tendo 288 milhões ativos. No facebook o número de usuários ativos foi de 1,44 bilhões, número que o Twitter critica a métrica utilizada pelo Facebook.

Algoritmos são utilizados para prever o sucesso de filmes em Hollywood, como o pesquisador Bernardo Huberman, da HP Labs, através do número de *tweets*, ele prevê como um produto vai se sair no mercado, no filme Querido John, previu uma arrecadação de US$ 30,5 milhões na primeira semana, o filme arrecadou US$ 30,7 milhões (BERNARDO, 2014).

O trabalho de (SANTOS, 2011) avaliou textos postados no Twitter sobre o Windows 7, investigando se o usuário apresenta ou não opiniões sobre o produto. Utilizou o método supervisionado de aprendizagem de máquina SVM para realizar a classificação binária desejada (positivo, negativo e neutro), constatou em seu resultado que o número de *tweets* coletados não foi satisfatório para se ter uma opinião concreta, o SVM possibilitou uma separação eficiente entre mensagens neutras e as que continham sentimentos, mas com as mensagens neutras separadas, haviam poucas mensagens positivas e negativas para a classificação binária.

[5] Disponível em: https://twitter.com
[6] Disponível em: http://www.facebook.com

2.4 Ferramentas para mineração de dados

Existem diversas ferramentas disponíveis no mercado, ferramentas genéricas de Inteligência Artificial (IA) ou de comunidades de estatísticas. Estas ferramentas são executadas separadamente do banco de dados, tendo assim um tempo gasto na exportação e importação dos dados, processamentos (pré e pós) e transformação dos dados; alguns autores defendem a conexão rígida entre a ferramenta de descoberta de conhecimento e o Sistema de Gerenciamento de Banco de Dados (SGBD) como desejável (DIAS, 2002).

2.4.1 WEKA

Uma destas ferramentas é o software Weka[7], que será utilizada para desenvolvimento do trabalho, siglas originadas do nome *Waikato Environment for Knowledge Analysis* (Waikato Ambiente para Análise do Conhecimento).

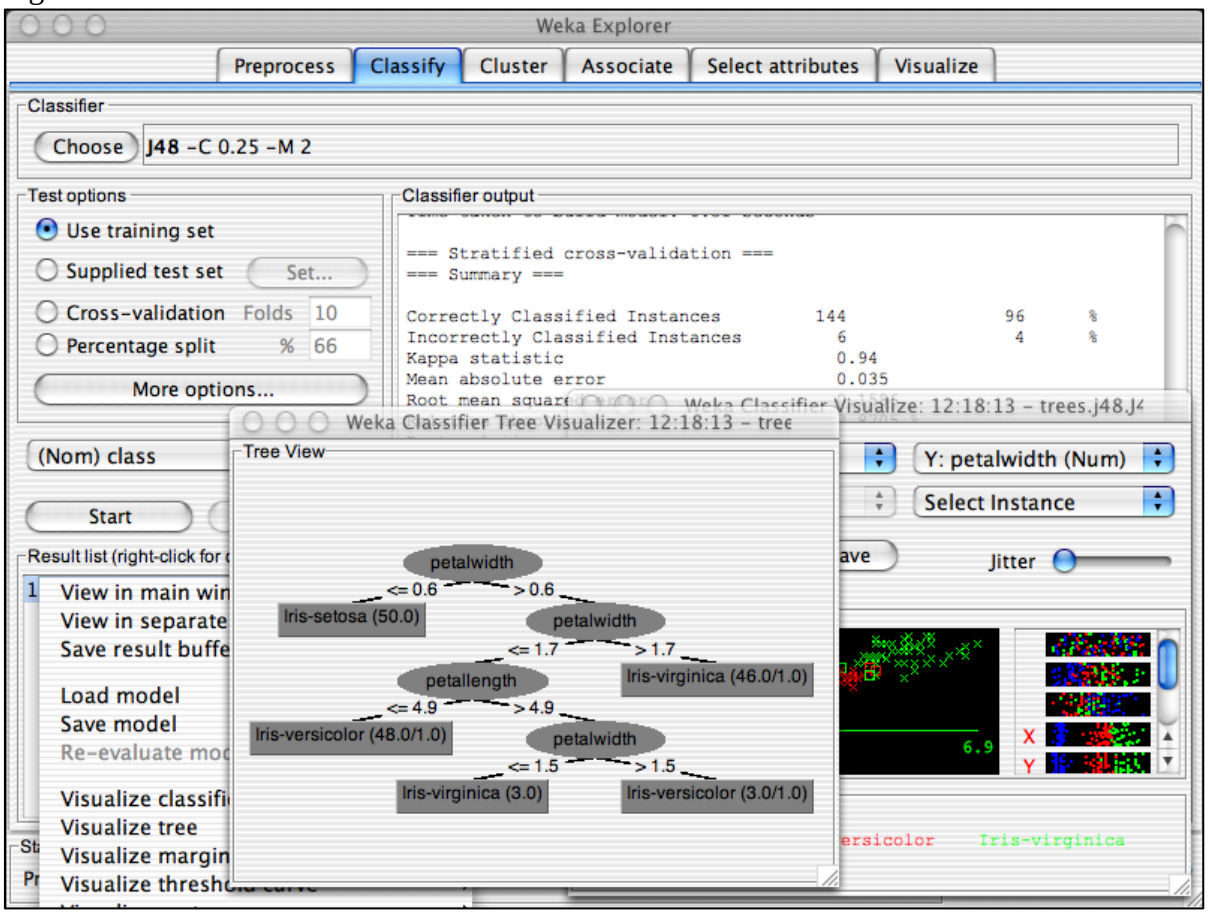
O pacote Weka está implementado na linguagem Java, pode rodar em diversas plataformas e com uma linguagem orientada a objetos como modularidade, polimorfismo, encapsulamento, reutilização de código dentro outros, sendo um software de domínio público (CIN, 2004).

O Weka não só fornece uma caixa de ferramentas de algoritmos de aprendizagem, mas também um painel interior, onde os pesquisadores implementam novos algoritmos, com suporte de infraestrutura para manipulação de dados e avaliação de esquemas. Tem ampla aceitação nos meios acadêmicos e ambientes de negócios (KHANALE, 2011).

O painel classificador permite configurar e executar um dos classificadores Weka no conjunto de dados atual. Há possibilidade de executar uma validação cruzada ou teste num conjunto de dados separados. Erros de classificação serão visualizados em outra tela pop-up. Caso o classificador gere uma árvore de decisão você pode visualizá-la graficamente em uma tela pop-up, como mostra a Figura 6.

[7] Disponível em: www.cs.waikato.ac.nz/ml/weka

Figura 6 - Painel Classificador do WEKA.

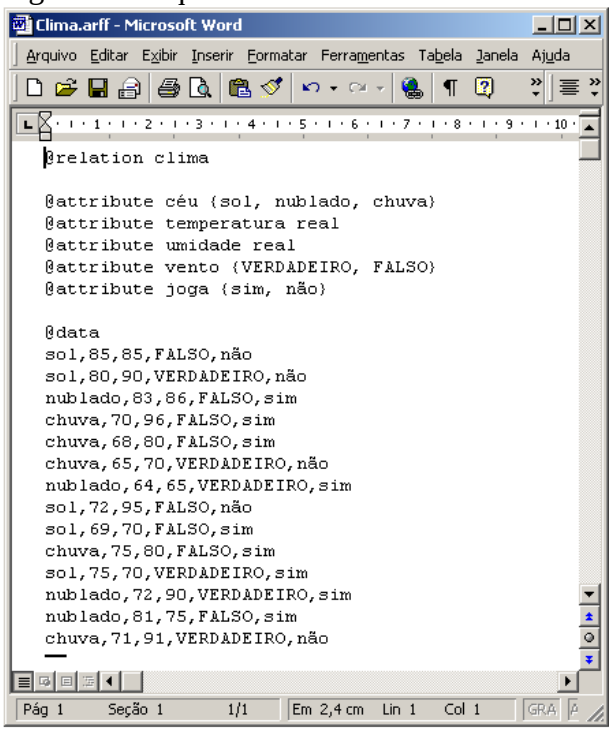


Fonte: WAIKATO (2015).

Antes de exportar os dados de treino para o Weka, deve-se converter para o formato ARFF, em duas partes. Na primeira parte se tem uma lista com todos os atributos, onde definimos o tipo de atributo ou os valores que eles podem representar. Na segunda parte consta os registros que serão minerados com o valor dos atributos para cada instancia separado por vírgula, caso não tenha algum registro, informa-se “?” (CIN, 2004).

A Figura 7 mostra um exemplo de arquivo no formato para importação na aplicação WEKA.

Figura 7 - Arquivo no formato ARFF.



Fonte: CIN (2004).

2.4.2 Outras ferramentas

A ferramenta R-Project[8], é uma linguagem e ambiente de gráficos e estatísticas computacionais, fornece uma ampla variedade de estatísticas (modelagem linear e não-linear, testes estatísticos, análise de séries temporais, classificação, *clustering* entre outros), ainda com técnicas gráficas, um dos pontos fortes da R-Project é a facilidade na produção de fórmulas, incluindo símbolos matemáticos.

Está disponível como software livre, compila e roda em amplas variedades de plataformas UNIX e os sistemas semelhantes como Windows e MaxOS.

A R tem em seu ambiente de manipulação de dados, conjunto de operadores para cálculos em tabelas e matrizes, na análise de dados contém uma grande coleção integrada e coerente, gráficos para análise de dados e visualização na tela ou cópia impressa.

É uma suíte integrada de instalações de software para manipulação de dados, cálculo e exibição gráfica, com manipulação de dados eficaz de armazenamento fácil; um conjunto de operadores para cálculos em tabelas, ou matrizes; ferramentas intermediárias para análise de dados; gráficos para análise de dados em visualização, na tela ou impressão; linguagem de

[8]Disponível em: www.r-project.org

programação simples e eficaz, loops, funções recursivas definidas pelo usuário e recursos de entrada e saída.

A ferramenta RapidMiner[9] desenvolvida por cientistas de dados para cientistas de dados, análise de negócios e desenvolvedores. Permite que usuários tenham acesso a todos os dados de todos os ambientes, fornecendo uma vantagem livre de código e conhecimento de milhares de usuários no mundo, tendo solução abrangente de análise de dados, fornecendo análises preditivas precisas, textos detalhados na mineração de dados.

Com a ferramenta RapidMiner Studio possui uma interface gráfica para processos analíticos. Carregamento de dados aberto e extensível, transformações de dados, modelagem de dados, métodos de visualização de dados com acesso a uma lista detalhada de fontes de dados, por exemplo: Excel, Access, Oracle, IBM DB2, Microsoft SQL, MySQL, Postgres, SPSS, texto arquivos entre outros.

Inclui aceleradores exclusivos para análise de sentimento, manutenção preditiva e marketing direto. Realizar anotações de fluxo de trabalho, colaborando com sua equipe ou toda a empresa com facilidade. Executado em todos os principais sistemas e plataformas operacionais, por exemplo: Windows, Linux, MacOS.

A ferramenta está disponível para ser baixada, possui um tempo limite de uso *free*, ao termino deste tempo o usuário precisa adquirir uma licença.

Demonstra ser uma ferramenta robusta, logo ao iniciar a aplicação o sistema convida o usuário a participar de um tutorial, tutorial que auxiliará na construção de uma análise de dados.

[9] Disponível em: rapidminer.com

3 TRABALHOS RELACIONADOS

Neste trabalho de conclusão foram utilizadas ferramentas de coletas desenvolvidas por alunos de universidades diferentes, trabalhos estes de conclusão da UNIVATES pelo acadêmico Bruno Edgar Fuhr em “Desenvolvimento de uma ferramenta de coleta e armazenamento de dados para Big Data” (FUHR, 2014) e da Universidade de Santa Cruz do Sul (UNISC) pelo acadêmico Gabriel Merten Bulsing em “Ferramenta para extração de dados semiestruturados para carga de um Big Data” (BULSING, 2013), outro acadêmico Roberto Antonio Schuster Filho em “Adaptação Temporal e Qualitativa sobre Mecanismos de Clipagem Eletrônica” (SCHUSTER FILHO, 2013). Outro trabalho relevante é a análise de resultado do acadêmico Vinícius Fernandes Schmitt em “Uma análise comparativa de técnicas de aprendizagem de máquina para prever a popularidade de postagens no Facebook” (SCHMITT, 2013). Também foi considerado o estudo de caso desenvolvido por Leonardo Augusto Sápiras e Karin Becker com a “Identificação de aspectos de candidatos eleitorais em comentários de notícias” (SÁPIRAS; BECKER, [201-]). Houve estudos na ferramenta desenvolvida no trabalho do acadêmico Ivan Luis Suptitz da UNISC, em “Aplicação de Mineração de Texto com o apoio de Ontologias para extração de conhecimento em Bases de Dados Textuais” (SUPTITZ, 2013).

3.1 Ferramenta de coleta de dados Mr. Crawler

No trabalho de (FUHR, 2014) foi desenvolvida uma ferramenta de coleta e armazenamento de dados chama Mr. Crawler, utilizando-se de coletores de dados (*crawlers*). Coletores que formam banco de dados com as informações obtidas, o Big Data. Compreendendo os princípios de armazenamento em banco de dados NoSQL, propondo uma arquitetura de coleta, processamento e análise de grande volume de dados.

Os dados coletados do Twitter virão dos *crawlers*, dados esses armazenados em big data MongoDB, banco de dados orientado a documentos substituindo o conceito de "linha" dos bancos relacionais, por um modelo flexível: o "documento". Documentos do MongoDB são do tipo JSON (*JavaScript Object Notation*), demonstração simples de dados, compreensível para homens e fácil interpretação para máquinas.

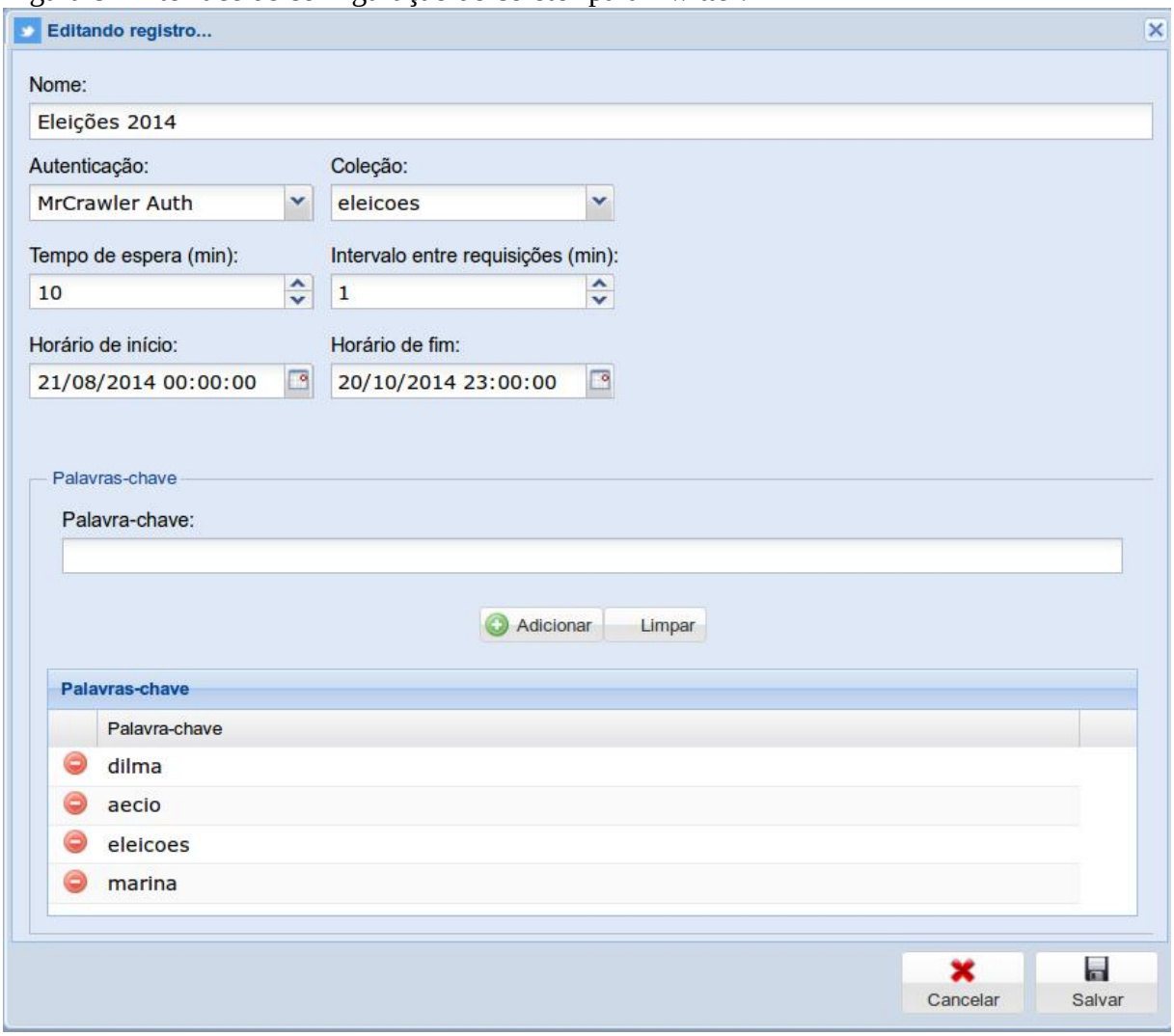
Através dos coletores e interface de configuração, foi implementada uma estrutura que coleta e armazena no MongoDB dados provenientes de diversas fontes, sendo assim possível ser usado um sistema de mineração de dados.

O módulo de coleta consulta as fontes de dados e armazena os dados obtidos, conforme configurado pelo usuário, existem 3 fontes de dados: Twitter, Facebook e Google Plus. Existe um processo que é executado em segundo plano, que controla as coletas paralelamente. As configurações dos coletores é feita a partir de uma interface gráfica, para gerenciamento da coleta de dados, podendo manter também o cadastro de servidores, banco de dados e coleções MongoDB. Com intuição de ser possível acessá-la em qualquer lugar com *internet*, a interface da ferramenta é do tipo web, sendo acessada através de um navegador, por exemplo: Mozilla Firefox ou o Google Chrome, sendo necessária a sua hospedagem em uma servidor de páginas.

Para a interface gráfica foi utilizada a biblioteca javascript ExtJs, executar operações no lado servidor foi utilizada a linguagem de programação PHP, para o armazenamento das informações de configuração dos coletores utilizou-se o SGBD PostgreSQL.

A Figura 8 mostra a tela de configuração dos coletores de dados para o Twitter, como o nome do coletor, autenticação utilizada, coleção onde serão armazenados os dados coletados, tempo de resposta à uma requisição à API, o tempo entre as requisições de início e fim de funcionamento do coletor e abaixo as palavras-chave que serão utilizadas para filtrar os dados do Twitter.

Figura 8 - Interface de configuração do coletor para Twitter.



Fonte: FUHR (2014).

3.2 Ferramenta Ontoclipping

No trabalho de (BULSING, 2013) é desenvolvida uma ferramenta que realize a extração de dados de diversas fontes da web, armazenando-as em um banco de dados Big Data, com computação distribuída e um framework Hadoop em Java que permite trabalhar com milhares de nodos e volume de dados na escala de *petabytes*. As informações extraídas pelo motor de busca serão armazenadas em um banco de dados NoSQL (Apache Cassandra).

Com a ideia de facilitar as buscas, (BULSING, 2013) utilizou estruturas de dados que visam auxiliar nas consultas, as Ontologias e as técnicas de Recuperação de Informação, podendo modelar, catalogar e indexar os termos específicos na área do conhecimento, tornando mais precisa a consulta.

A Figura 9 mostra a tela de configuração do motor de busca para execução do *crawlers*, na opção nível, através dessa opção é definido a quantidade de níveis em que os *crawlers* irão buscar a partir dos sites sementes; a opção Domínio com ela ativada o *crawler* visita e coleta somente *sub-links* das páginas que pertençam ao mesmo domínio da página semente, por exemplo: excluindo as propagandas; a opção *All* Texto sendo ativada realiza a coleta de todo o texto da página independente de qual *tag*HTML ele estiver incluso; No Indexar possibilita que a ferramenta já realize a indexação logo após realizar a extração dos dados; a opção *Print Console* sendo habilitada permite visualizar no console do Eclipse as URLs que estão sendo percorridas pelo motor de busca.

Figura 9 - Configurações do motor de busca.

Fonte: BULSING (2013).

Nas opções de execução é possível fazer uma consulta por termos nos dados já coletados pelo *crawlers*, deixando desmarcado o *checkbox* "Executar Crawler". Deixando marcado o motor de busca iniciará uma nova coleta de informações baseadas nos sites sementes, não havendo duplicidade de dados.

Java foi a linguagem de programação utilizada, para a interface gráfica foi utilizado a API JavaFX 2, para a indexação das buscas foi utilizado o Apache Lucene, o qual é uma biblioteca de indexação e busca textual escrita inteiramente em Java.

3.3 Clipagem Eletrônica

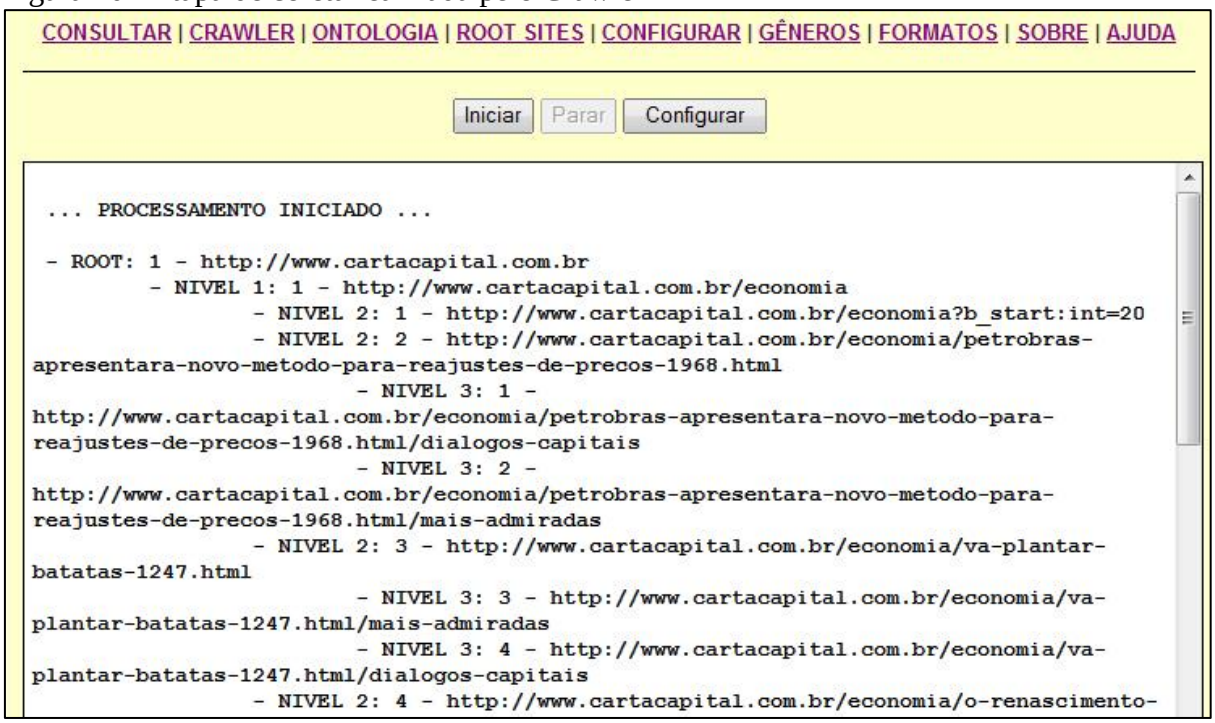
O trabalho de (SCHUSTER FILHO, 2013) evoluiu o software OntoClipping, implementando funcionalidades que atendem a clipagem e *Data Quality*, considerando fatores como visão do usuário, categorização, peso e abrangência das fontes de informação.

Conforme (SCHUSTER FILHO, 2013), a definição de *clipping* ou clipagem resume-se na busca e coleta de informações, atendendo a um interesse específico. A qualidade de dados (*Data Quality*) e seus efeitos em cada tipo de atividade estão mais críticos a cada dia,

dados eletrônicos, certo ponto são de melhor qualidade comparados aos de papel. Os processos que dão origem a esses dados, na maioria, estão fora do controle, ocasionando erros nos dados.

Na etapa do Crawler, é feita a coleta dos dados de páginas web, dos sites raízes pré-cadastrados em etapas anteriores, o conteúdo é armazenado em um banco de dados relacional, caracterizando parte do processo de *clipping*. Durante a coleta são realizadas a Extração de Completude, verificando as características dos dados coletados como, título, data e outros, e a Extração de Gêneros e Formatos Jornalísticos, depende da etapa anterior, sendo o formato jornalístico identificado na página *web* na Extração de Completude, logo após sendo identificado o gênero, onde foram associados em etapas anteriores. No formato jornalístico é representado em uma linguagem não padronizada, sendo auxiliado nos termos equivalentes cadastrados na etapa de Configuração de Formatos, a Figura 10 mostra o crawler em execução.

Figura 10 - Etapa de coleta realizada pelo Crawler



Fonte: SCHUSTER FILHO (2013).

3.4 Análise de técnicas na Aprendizagem de Máquinas

No trabalho do (SCHMITT, 2013) foi feita uma avaliação do desempenho de três classificadores de aprendizado de máquina supervisionados, em uma página de fans do Facebook, na tarefa de classificar a popularidade do conteúdo textual das postagens.

Os algoritmos de classificação escolhidos foram o *naive Bayes*, Support Vector Machines e o C4.5, pelo motivo de serem os mais influentes utilizados pela comunidade de pesquisa na área de classificação.

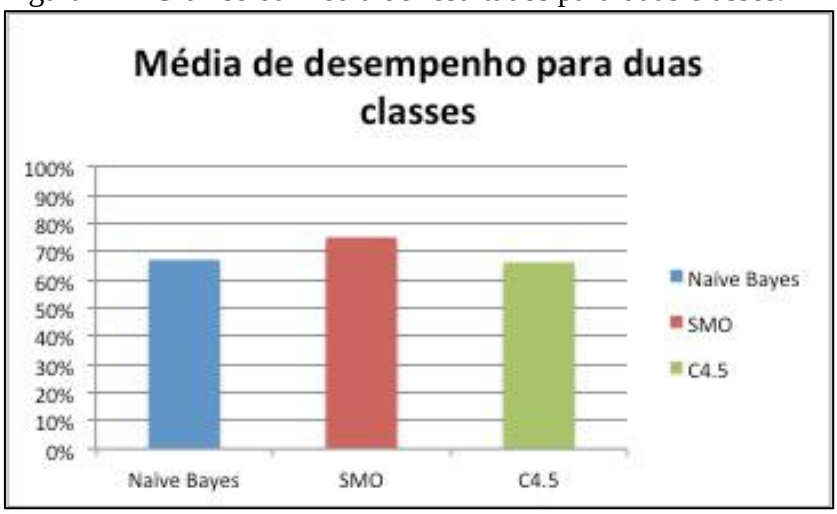
Para a classificação das postagens, levou-se em consideração o número de curtidas ou likes de cada postagem. A classificação foi feita considerando duas e três classes possíveis de popularidade. Nos testes considerando duas classes possíveis (“boa” e “ruim”), foram divididas em dois grupos aproximadamente 50% por classe. Considerando três classes (“boa”, “média” e “ruim”), foram divididas em três grupos com aproximadamente 33%.

Para realizar a classificação dos dados, (SCHMITT, 2013) realizou o pré-processamento para construir os dados de treino, processo esse que neste trabalho será desempenhado pela ferramenta a ser desenvolvida.

Conforme (SCHMITT, 2013) percebeu analisando os resultados que se obtêm melhores resultados quando se considera apenas duas classes comparado a três classes, uma hipótese mais provável da obtenção de piores resultados com três classes, é que a classe “média” traga muitos ruídos no modelo.

Considerando as respectivas acurácias dos modelos gerados, como mostra a Figura 11 o algoritmo SMO apresentou as melhores médias, tanto para duas e três classes, assim como a melhor combinação dentro dos testes realizados, com 80,5% de acurácia.

Figura 11 - Gráfico da média de resultados para duas classes.



Fonte: SCHMITT (2013).

3.5 Identificação de aspectos na mineração de opinião

O estudo de caso de Sápiras; Becker ([201-]) buscou desenvolver técnicas para realizar mineração de opinião em nível de aspecto em fontes de dados menos estruturadas, como comentários, *tweets*, *blogs* ou jornais. Opiniões expressas em relação a candidatos a eleições.

O foco específico é no aspecto da entidade, no real alvo da opinião. Saúde e educação são os dois aspectos do plano eleitoral considerados. Através disso, apresentar um conjunto de experimentos, desenvolvidos na busca da identificação das técnicas mais adequadas, e sobre quais documentos aplicar.

Foram utilizadas as técnicas Palavras Sementes, Expected Mutual Information Measure (EMIM), Phi-squared e Indexação Semântica Latente (LSI) para identificar os aspectos nos comentários sobre notícias políticas, bem como a avaliação dos resultados obtidos. Na avaliação dos resultados, foram utilizadas métricas de Precisão, Revocação, F-score e Acurácia.

Baseado no F-score, a abordagem com melhor avaliação sobre Saúde foi a phi-squared, na Educação o melhor resultado foi com a EMIM. Através dos resultados desses experimentos, agregar técnicas ao presente trabalho, pois o estudo de caso verificou que é valido considerar apenas os comentários.

3.6 Mineração de Texto com apoio de Ontologias

No trabalho do SUPTITZ (2013) foi desenvolvida uma ferramenta que recupera informações do banco de dados dos serviços prestados por uma empresa de TI com o apoio de ontologias, servidor de apoio a consultores, técnicos e desenvolvedores que trabalham com suporte a usuários.

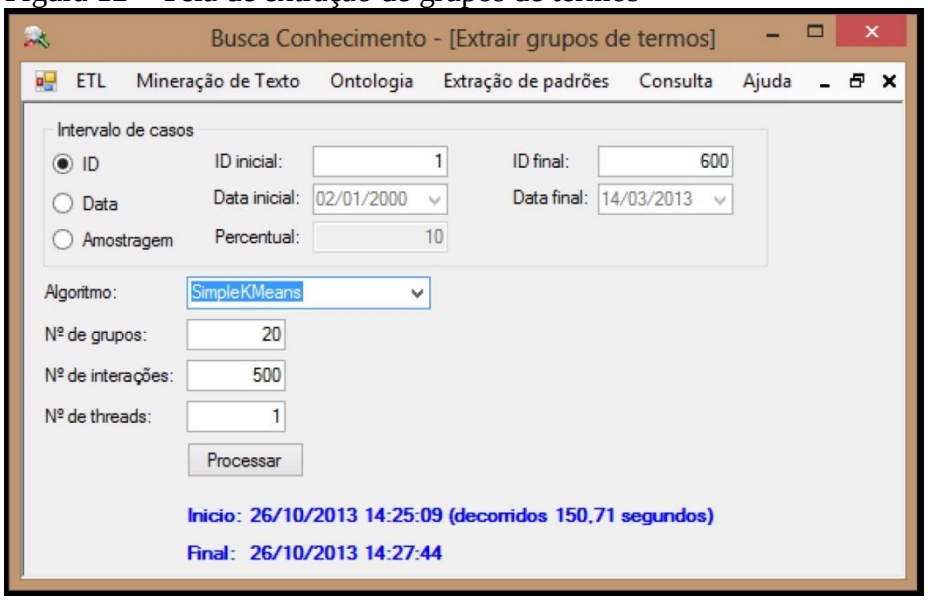
A ferramenta possibilita a manutenção das conexões às fontes dos dados que devem ser extraídos e processados, conexões estas utilizadas nos processo de importação dos dados que serão usados na mineração de textos.

Conta com o processo de pré-processamento, onde ocorre o tratamento do texto, removendo comandos HTML, filtro de caracteres permitidos, remoção de *stopwords*, corrigir ortografias e radicalização. Desenvolveu o processo de indexação de termos, que gera uma estrutura de índices que serão utilizadas para consulta e geração de padrões.

A manutenção manual de ontologias, conceitos vinculados, de acordo com sua hierarquia dentro da estrutura definida com base nos relacionamentos estabelecidos entre os conceitos. Conforme SUPTITZ (2013), para o apoio na classificação e recuperação de textos, existem trabalhos bem sucedidos na área de mineração de texto mostrando que o conceito de ontologia agrega em muito na obtenção de resultados.

A Figura 12 mostra a tela de extração de grupos de termos, o diferencial nos processos da ferramenta é apoiar o especialista na tarefa de criar e manter a ontologia, com a utilização da ferramenta Weka e foi realizada a construção de um software de apoio com interação com a ferramenta desenvolvida, nesta tela só é mostrada a interface da interação.

Figura 12 – Tela de extração de grupos de termos



Fonte: SUPTITZ (2013).

Neste processo foram utilizados algoritmos de agrupamento como o "*SimpleKMeans*", "*xMeans*" e "*HierarchicalClusterer*". A ferramenta conta com uma interface de consulta onde o usuário informa os termos da consulta que pretende submeter, seguido pelos resultados que serão achados, ordenados por relevância conforme cálculo baseado em referência de trabalhos pesquisados. Essa interface de consulta é aplicada todas as funcionalidades importantes do ponto de vista prático do objetivo do trabalho.

Para fins, realizou dois estudos de casos distintos, no primeiro utilizou uma coleção de documentos previamente catalogados com o objetivo de gerar estatísticas com as métricas *precision* e *recall*, verificando o comportamento da aplicação em um ambiente distinto do problema proposto; no segundo tratou-se com dados de chamados técnicos e ordens de serviços de uma empresa de TI, configurando o ambiente do problema proposto.

4 MATERIAIS E MÉTODOS

Após apresentar os principais conceitos relacionados aos objetivos deste trabalho, neste capítulo será apresentada a metodologia, técnicas, documentos e estruturas que farão parte da proposta do trabalho.

4.1 Metodologia

Na fundamentação do trabalho será utilizado o método estruturalista, o método parte da investigação de um fenômeno concreto, em seguida para o nível abstrato, pelas normas de um modelo que represente o objeto de estudo, retornando por fim ao concreto, desta vez com uma realidade estruturada. A linguagem abstrata se torna indispensável para se fazer comparações nas experiências à primeira vista irredutíveis, em outras palavras, que poderiam não ter explicação. O método estruturalista caminha do concreto para o abstrato ou vice-versa (LAKATOS, 2010).

Após realizar os requisitos principais para a execução do projeto, iniciou-se o desenvolvimento da ferramenta. Ao final desta etapa os dados são coletados de avaliações feitas por usuários em páginas *web* de lojas eletrônicas, processo de coleta realizado manualmente pelo autor e durante esse processo a ontologia (aspecto) e polaridade (sentimento) são atribuídas aos textos.

Com os dados coletados, partiu-se para o agrupamento e classificação dos mesmos. Os dados são carregados e filtrados, algoritmo filtro StringToWordVector escolhido para o trabalho, como já comentado na seção 2.2.2, os atributos receberão pesos numéricos conforme frequência nas frases. Pronto isso foram definidos os algoritmos de classificação, passando assim pelo processo de classificação, avaliação e visualização dos resultados.

4.2 Tecnologias Utilizadas

No desenvolvimento da ferramenta que foi projetada neste trabalho, foram utilizadas diferentes tecnologias de acordo a necessidade da solução do objetivo. Na aplicação dos algoritmos de classificação e avaliação dos dados, será utilizada a API Java do Weka, devido à facilidade no acesso e suporte ao conteúdo.

A ferramenta do trabalho será desenvolvida através do Microsoft Visual Studio 2008, aplicação escolhida pela facilidade que o autor já possui com suas ferramentas, linguagem Visual Basic.

Para fazer a ponte, importando as bibliotecas utilizadas pelo Weka, da linguagem Java para o .NET, etapa que o arquivo *.jar é convertido para *.dll, foi utilizada a Java Virtual Machine (JVM) a IKVM.NET[10], utilizando-a na forma estática onde permiti que o código Java seja utilizado por aplicações .NET.

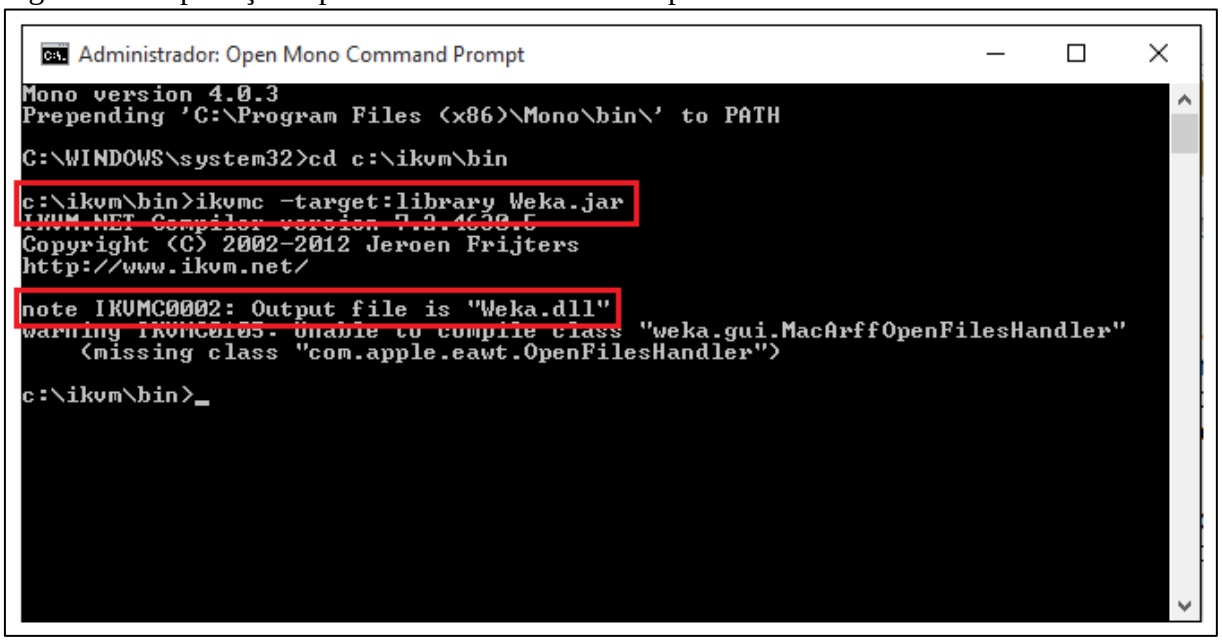
A versão Visual Studio 2008 foi escolhida devido a compatibilidade com a IKVM.NET. A IKVM.NET fornece as tecnologias relacionadas ao VM para a tradução byte-code e verificação, carregamento de classes. É dependente do projeto OpenJDK para implementação das bibliotecas do JDK até versão 7 onde houve êxito de compilação (IKVM, 2015), foi tentado utilizar o JDK 8 e não houve êxito na compatibilidade.

Para utilização dos algoritmos do Weka no desenvolvimento do .NET, foi compilada a biblioteca Weka.jar, encontrada na raiz de instalação da ferramenta Weka.

A compilação é executada através da plataforma Mono[11], segundo MONO (2015) a plataformo Mono é um software projetado para criação de aplicações multi-plataformas. Através do Mono é chamado o ikvmc, que realiza a conversão *.jar para *.dll, como mostra a Figura 13.

[10] http://www.ikvm.net/
[11] http://www.mono-project.com/

Figura 13 – Aplicação Open Mono Command Prompt



Fonte: Elaborado pelo Autor (2015).

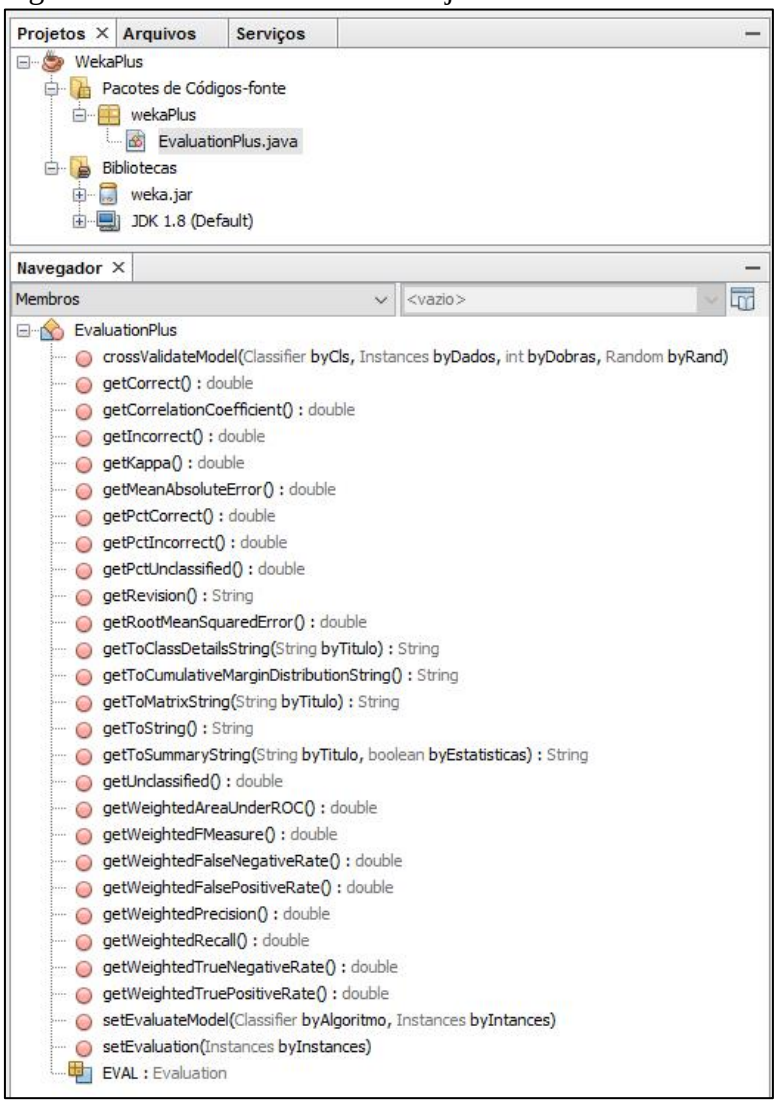
O arquivo Weka.dll foi carregado nas referencias do projeto AMD. Através disso a maioria das bibliotecas do Weka foram acessadas, mas não havendo êxito com a biblioteca "weka.classifiers.Evaluation", que é responsável pela avaliação dos dados classificados. Foram adquiridos outros arquivos Weka.jar, mas mantinha a mesma situação ou haviam mais bibliotecas sem acesso.

A avaliação dos dados é importante para a busca e amostra das classificações dos dados treino, tendo essa necessidade foi criado um projeto WekaPlus na linguagem Java. Neste projeto foi criada a biblioteca chamada EvaluationPlus.jar e adicionada a biblioteca Weka.jar, neste cenário foi possível acessar os recursos do Weka, tendo êxito na importação da biblioteca "weka.classifiers.Evaluation". Utilizado a ferramenta NetBeans IDE[12] como apoio no desenvolvimento do EvaluationPlus.jar, foram criadas as principais funções para utilização conforme necessário para o projeto AMD.

Na Figura 14 é mostra a estrutura criada na classe EvaluationPlus para suprir as necessidades da ferramenta AMD.

[12] https://netbeans.org/

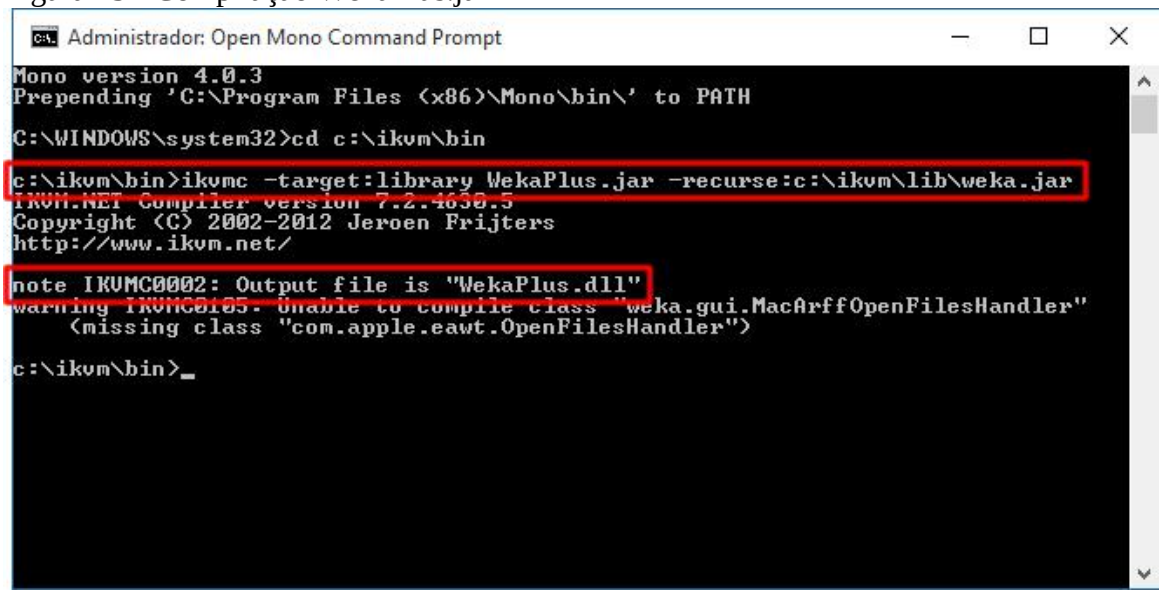
Figura 14 - Classe EvaluationPlus.java



Fonte: Elaborado pelo autor (2015).

Foi utilizado do IKVM.NET para compilar o EvaluationPlus.jar para EvaluationPlus.dll, assim conseguindo realizar as avaliações dos algoritmos de classificação. Como o projeto WekaPlus utiliza da biblioteca Weka.jar, no comando de compilação do ikvmc foi necessário adicionar a opção “-recurse”, onde é informado o diretório e arquivo que apoiará nos recursos da compilação, com destaca a Figura 15.

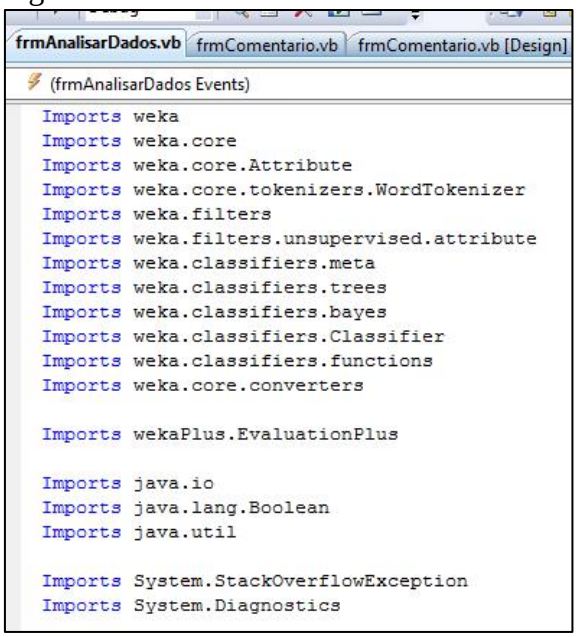
Figura 15 - Compilação WekaPlus.jar



Fonte: Elaborado pelo autor (2015).

Na Figura 16 podem ser conferidas as importações necessárias e possíveis das bibliotecas Weka.dll e WekaPlus.dll.

Figura 16 - Bibliotecas Weka e WekaPlus

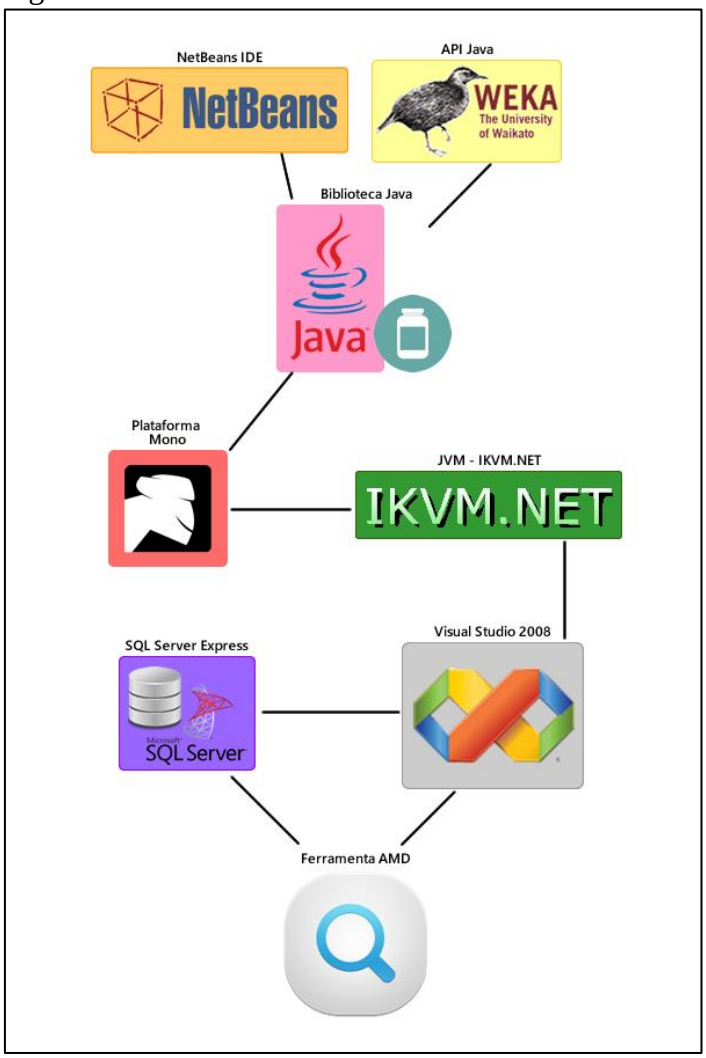


Fonte: Elaborado pelo autor (2015).

O SGBD do banco de dados utilizado foi o Microsoft SQL Server Express Edition, onde serão gravadas as configurações dos cenários do usuário, cadastro de ontologias e polaridades, comentários entre outros.

A estrutura das ferramentas utilizadas pode ser conferida conforme Figura 17.

Figura 17 - Ferramentas Utilizadas



Fonte: Elaborado pelo Autor (2015).

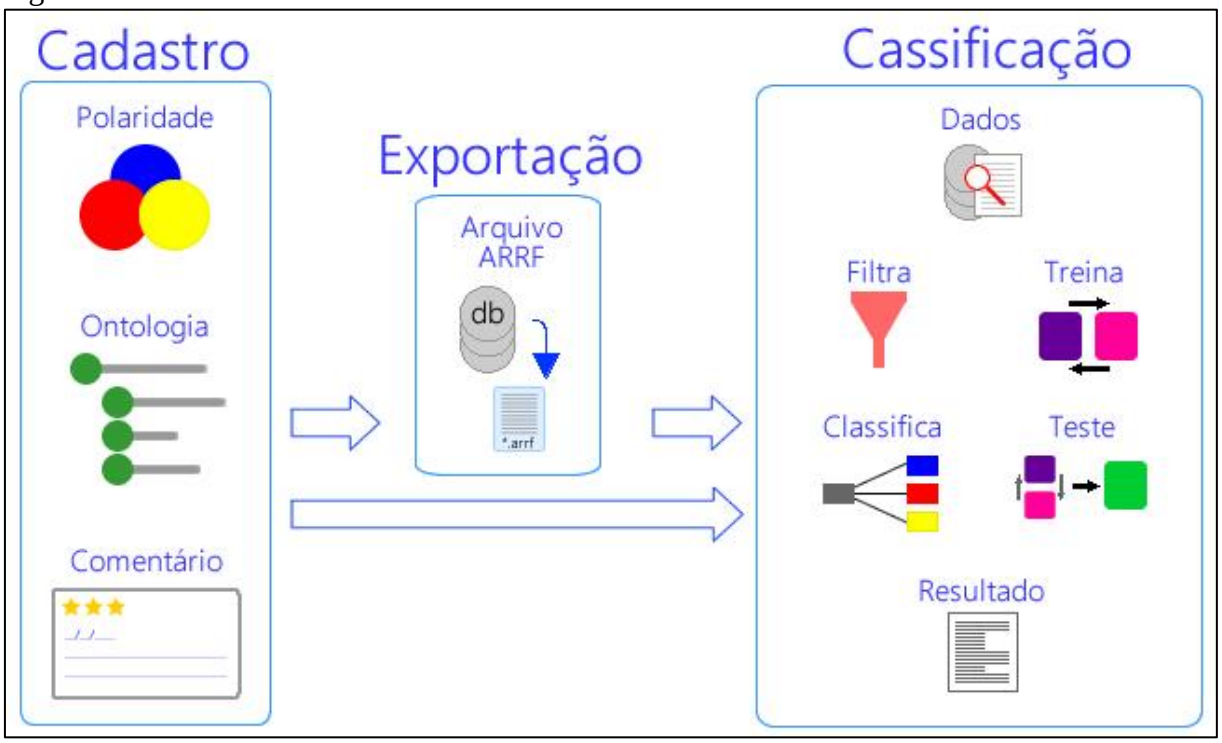
5 RESULTADOS E DISCUSSÃO

Nesta seção, será apresentada a ferramenta desenvolvida, com fundamento nas seções anteriores. Seguindo pelo capítulo de desenvolvimento do sistema, com seus módulos, com a modelagem do banco, telas, testes, resultados obtidos e comentados.

5.1 Desenvolvimento da Ferramenta

Como mostra na Figura 18, a ferramenta será composta por três módulos ou recursos: cadastros, exportação e classificação.

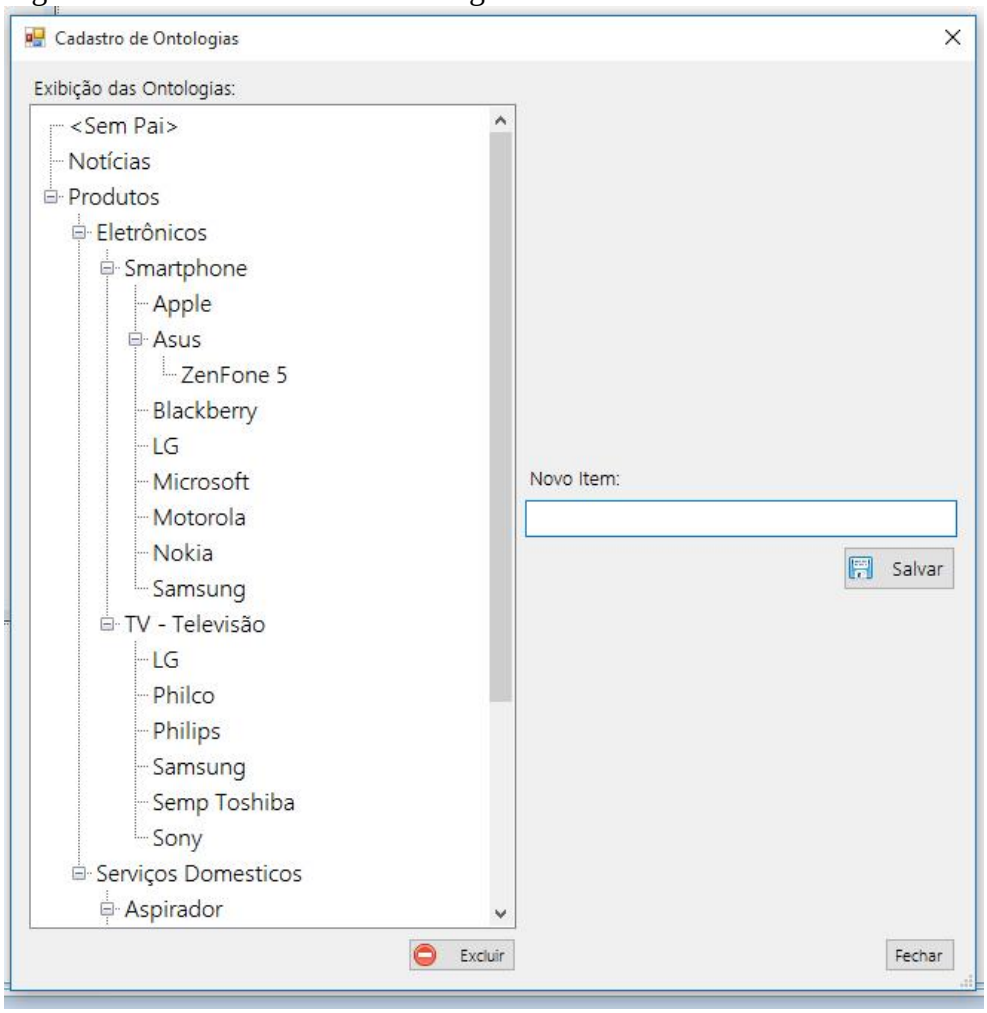
Figura 18 - Módulos/Recursos da Ferramenta



Fonte: Elaborado pelo autor (2015).

Na seção configuração/cadastro há possibilidade de configurar o acesso ao banco de dados e cadastrar as ontologias (aspectos) como mostra a Figura 19. O usuário poderá cadastrar diversas ontologias pertinentes ao seu cenário de pesquisa, através desta estrutura hierárquica será dado apoio a classificação e recuperação de textos, como comentado na seção 3.6 por SUPTITZ (2013). A tela permite a inclusão e exclusão dos registros e contém uma listagem hierárquica das ontologias. Na forma que foi estruturada as ontologias agregam diversos termos chaves aos *crawlers*, conforme exemplo da Figura 19. Selecionando a ontologia "Eletrônicos", o sistema pode gerar um conjunto de termos chaves conforme nível hierárquico, por exemplo: Smartphone / TV – Televisão / Apple / Asus / ZenFone 5 / Blackberry entre outros.

Figura 19 - Tela Cadastro de Ontologias



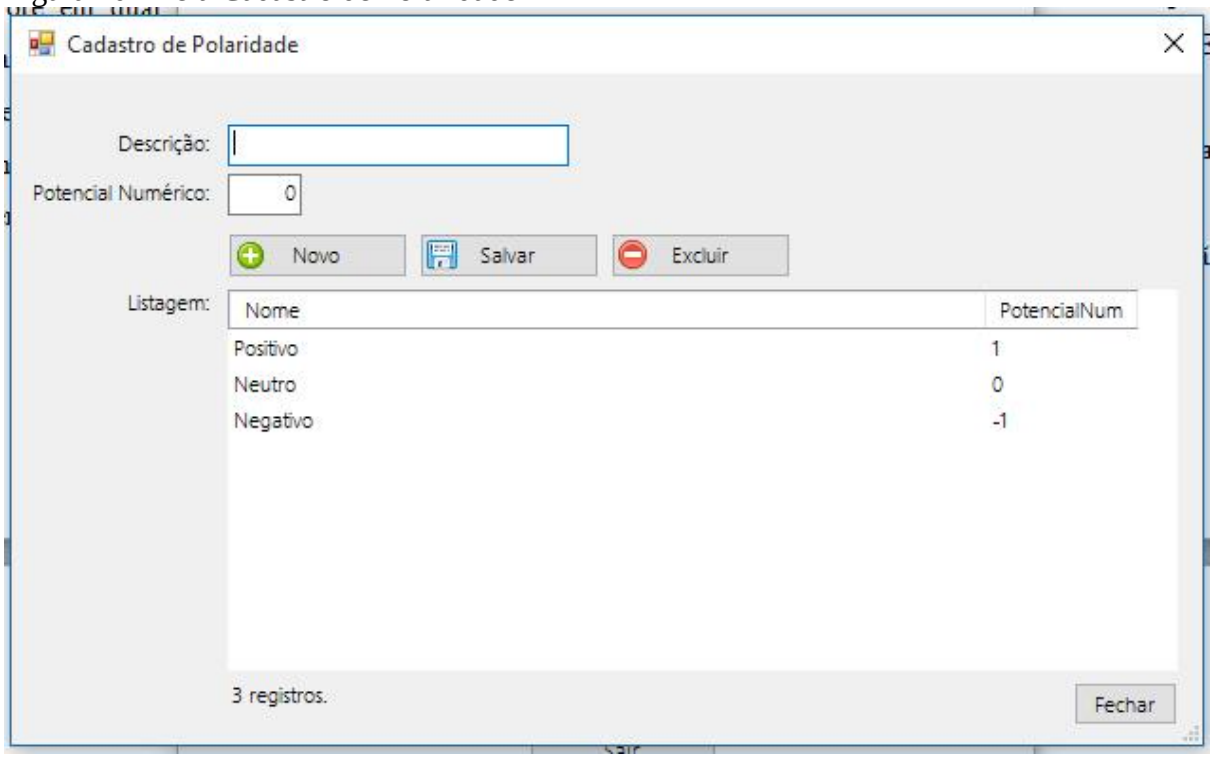
Fonte: Elaborado pelo autor (2015).

Na Figura 20 é mostrada a tela de cadastro das polaridades, polaridades estas que serão atribuídas aos textos juntamente com as ontologias, auxiliando na definição dos sentimentos como comentado na seção 2.3 por CARVALHO (2014). O cadastro possui três

botões para manutenção dos dados, campos Nome e Potencial Numérico, por exemplo: Ótimo: 2, Bom: 1, Neutro: 0, Ruim: -1, Péssimo: -2.

A polaridade deve ser definida manualmente para que a ferramenta de mineração de dados tenha um embasamento através da classe atribuída ao texto, assim esses exemplos serão usados na comparação com os dados que estão sendo minerados.

Figura 20 - Tela Cadastro de Polaridade



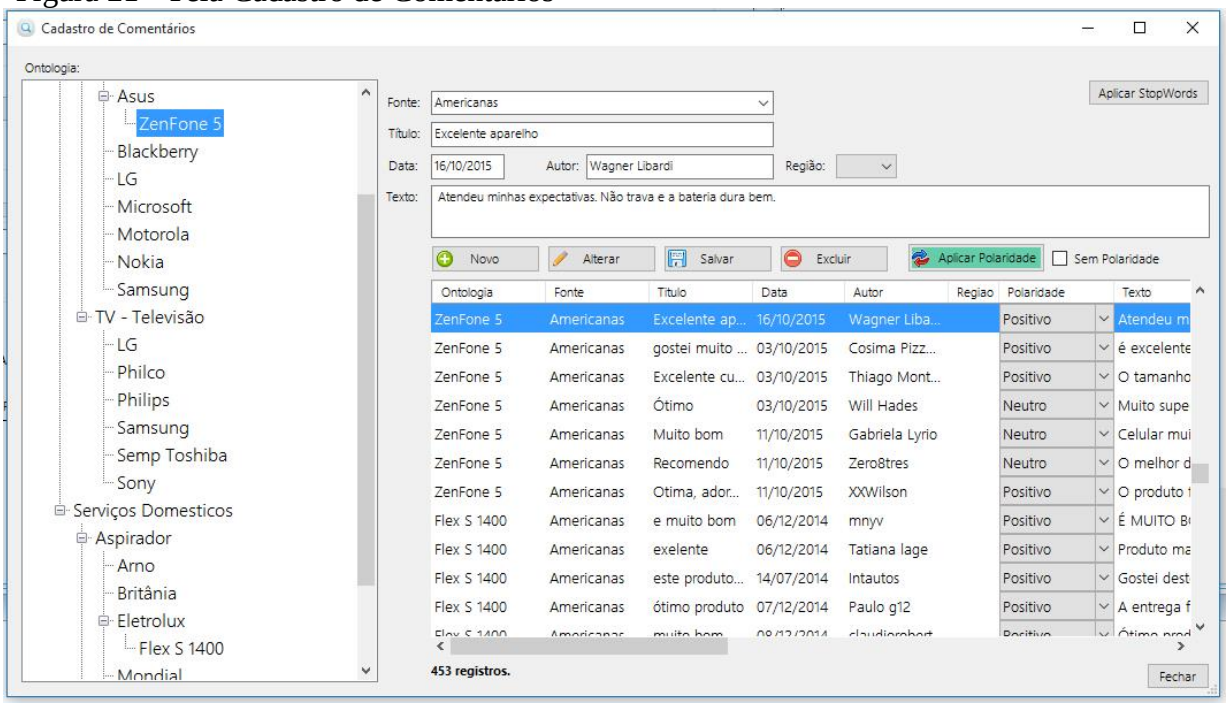
Fonte: Elaborado pelo autor (2015).

Uma tela de grande importância para a ferramenta é a dos comentários coletados, como mostra a Figura 21. Nesta tela o usuário realiza a manutenção dos dados adquiridos, define a ontologia, a polaridade, a fonte e outros campos como Título, Data, Autor, Região e Texto conforme coleta. Existem quatro botões para a sua manutenção, na opção “Aplicar StopWords” serve para remover as palavras que são irrelevantes, como comentado na seção 2.2.2, opção que será útil à classificação, este processo acessa um arquivo SW-PT.txt localizado na raiz da ferramenta, arquivo este carregado com 300 palavras em português, palavras adquiridas no site (LINGUATECA.PT, 2015). Utilizar o *stopwords* padrão do Weka torne-se inviável para o objetivo do trabalho, já que todas as palavras estão na língua inglesa.

A opção Aplicar Polaridade atribui polaridade selecionada ao texto, existe um combo na tabela com uma relação conforme polaridades cadastradas, como é um processo ardo ler todos os textos, analisar e atribuir um sentimento, foi analisado uma maneira de tornar

amigável esse processo de polarizar, o usuário informa a polaridade na coluna Polaridade, em seguida aplica-se ela ao texto clicando na opção Aplicar Polaridade, não há necessidade de cada texto polarizado clicar neste recurso, definem-se vários quando desejar aplica-se aos textos. Com o aumentar dos dados, a tabela pode se tornar grande, assim há uma opção (Sem Polaridade) para filtrar somente os textos que ainda não possuem polaridade definida, auxiliando na aplicação.

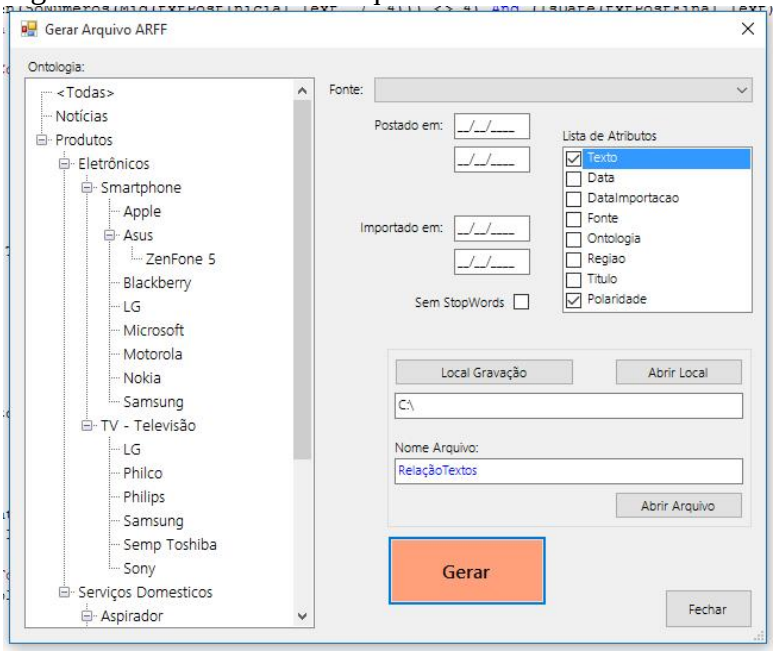
Figura 21 - Tela Cadastro de Comentários



Fonte: Elaborado pelo autor (2015).

Considerando que para aplicar os algoritmos do Weka é necessário gerar um arquivo no formato arff, arquivo comentado na seção 2.4.1, o modulo de exportação foi desenvolvido na tela Gerar Arquivo arff como mostra a Figura 22, o usuário poderá filtrar os dados de acordo com a sua necessidade de análise. À esquerda há a árvore das ontologias, caso queira uma relação de textos das ontologias descendentes coletadas do Smartphone, basta selecionar a ontologia Smartphone.

Figura 22 - Tela Gerador Arquivo arff



Fonte: Elaborado pelo autor (2015).

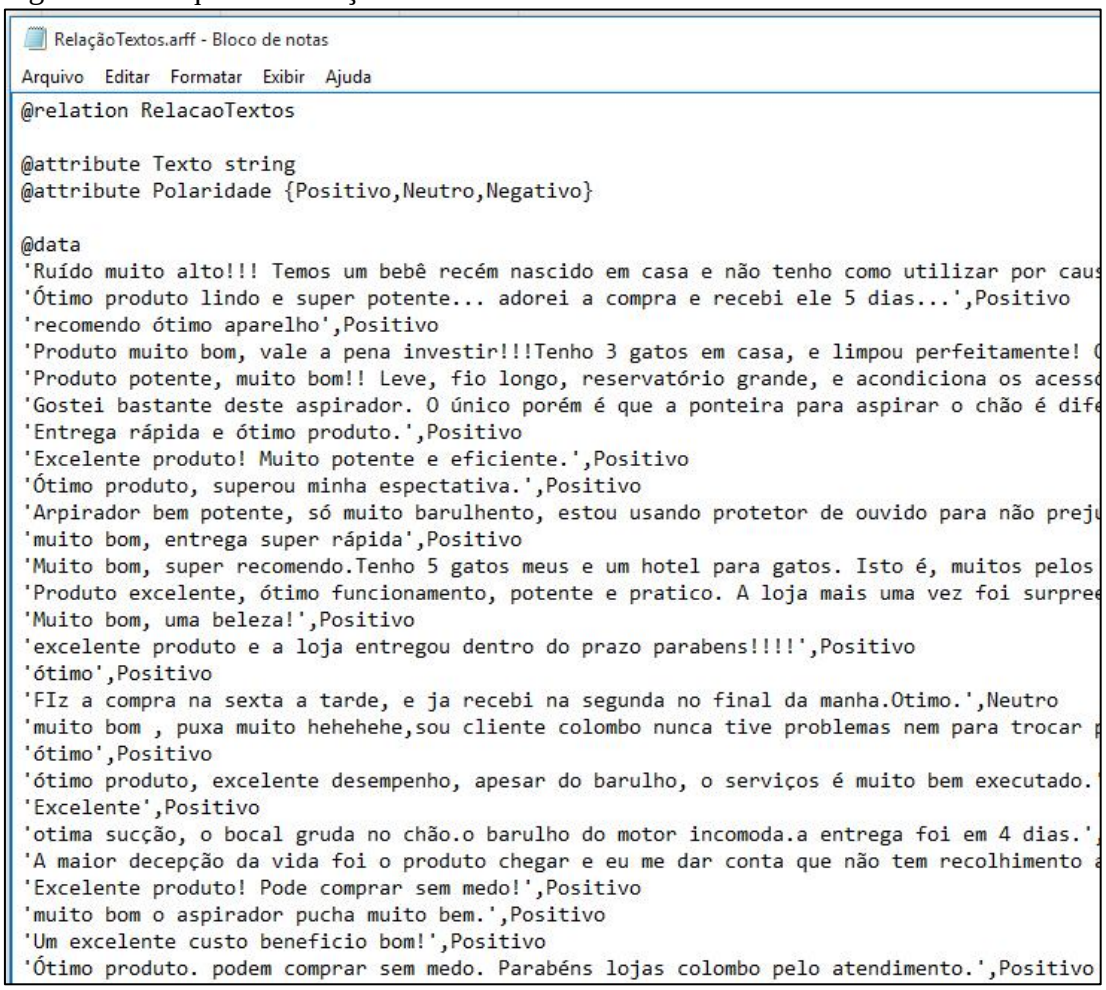
No lado direito da tela existem outros filtros como a Fonte dos textos, o período que foi postado os textos, o período que foi importado os textos. Há o filtro “Sem StopWords”, filtro que exporta para o arquivo o texto sem as palavras consideradas irrelevantes, as *stopwords*, para a classificação.

A Lista de Atributos é de seleção obrigatória, o usuário deve informar quais os atributos (colunas) deverão constar no arquivo *.arff, essa listagem é de acordo com as colunas que a tabela comentário possui.

Na seção da identificação do arquivo o usuário tem total liberdade do local de gravação e nome que será dado ao arquivo. Possui o botão Local Gravação onde é definido o local de gravação, o botão Abrir Local para facilitar no processo de consulta após geração e o botão Abrir Arquivo, para fins de conferência dos dados gerados. Esses dois últimos botões se tornam totalmente práticos na usabilidade do sistema.

E o botão Gerar onde serão gerados arquivos conforme filtros, a Figura 23 mostra o conteúdo gerado no arquivo *.arff, semelhante à Figura 7 da seção 2.4.1.

Figura 23 - Arquivo - Relação Textos.arff

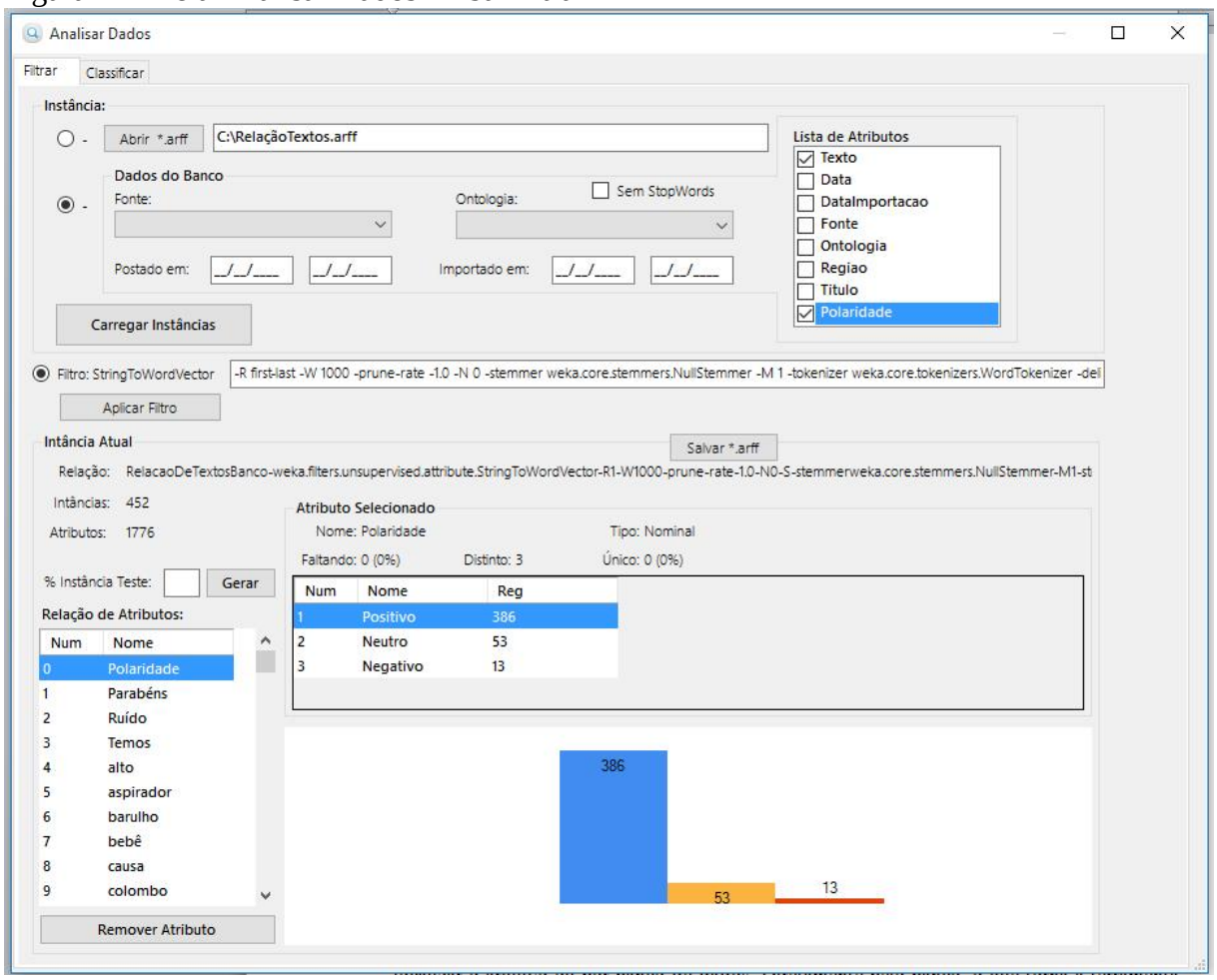


Fonte: Elaborado pelo autor (2015).

A tela foi elaborada de forma que o usuário tenha total flexibilidade nas formas de geração do arquivo. Houve o desenvolvimento da tela para realizarmos comparações das avaliações dos algoritmos também dentro da ferramenta Weka.

No modulo classificação está o principal da ferramenta AMD, a tela Analisar Dados como mostra a Figura 24.

Figura 24 - Tela Analisar Dados - Aba Filtrar



Fonte: Elaborado pelo autor (2015).

Na aba Filtrar ocorre às etapas de conhecimento do domínio e pré-processamento da Mineração de Dados, comentados na seção 2.2. O usuário efetuará a busca pelos dados, o conjunto de dados denominado Instância, há duas opções de busca, através de arquivos arff, onde há a opção de localizar o arquivo ou por banco de dados. Diretamente pelo banco, o que torna a ferramenta AMD um diferencial, o usuário não precisa entender de arquivos arff, de sua estrutura para conseguir classificar os dados adquiridos.

Foi criada essa segunda opção para facilitar a busca pelos dados, basicamente foram colocados os mesmos campos que constam na tela de geração do arquivo arff, acreditando ser o suficiente para suprir as necessidades para criação da instância.

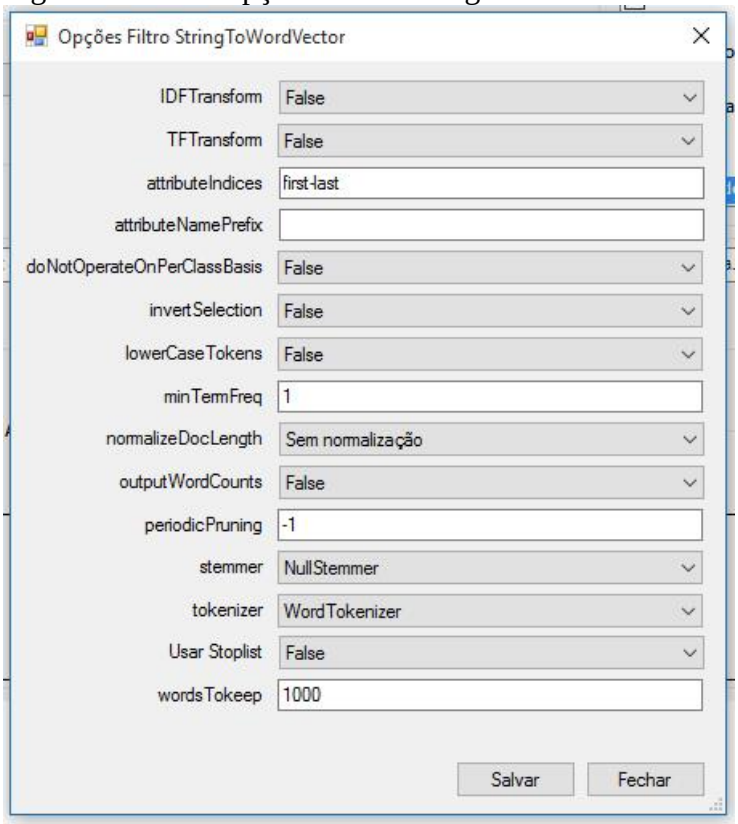
Após carregar a instância conforme opção do usuário, já se tem uma visualização dos dados carregados da instância, na seção Instância Atual. É informado ao usuário a relação, o número de instâncias (linha de dados) e número de atributos, no lado direito algumas informações de estatísticas da relação instância x atributo. Caso o usuário queira remover

atributos que não tenha importância para a análise dos dados há o botão Remover Atributo. Esta opção na ferramenta auxilia na etapa de pré-processamento da mineração dos dados, como comentado na seção 2.2.2, dando suporte ao analista no conjunto de dados, verificando valores válidos, preferências, restrições, evitando futuros problemas nos algoritmos.

O gráfico é mostrado quando o usuário selecionar um atributo do tipo Nominal, mostrando o número de registros conforme classes, conforme exemplo da Figura 21, o número de instâncias da classe positiva, neutro e negativa.

O filtro StringToWordVector comentado na seção 2.2.2 por WAIKATO (2015), possui diversas opções de filtragem, essas opções o usuário tem liberdade de redefini-las do padrão, através da tela Opções Filtro StringToWordVector, como mostra a Figura 25.

Figura 25 - Tela Opções Filtro StringToWordVector



Fonte: Elaborado pelo autor (2015).

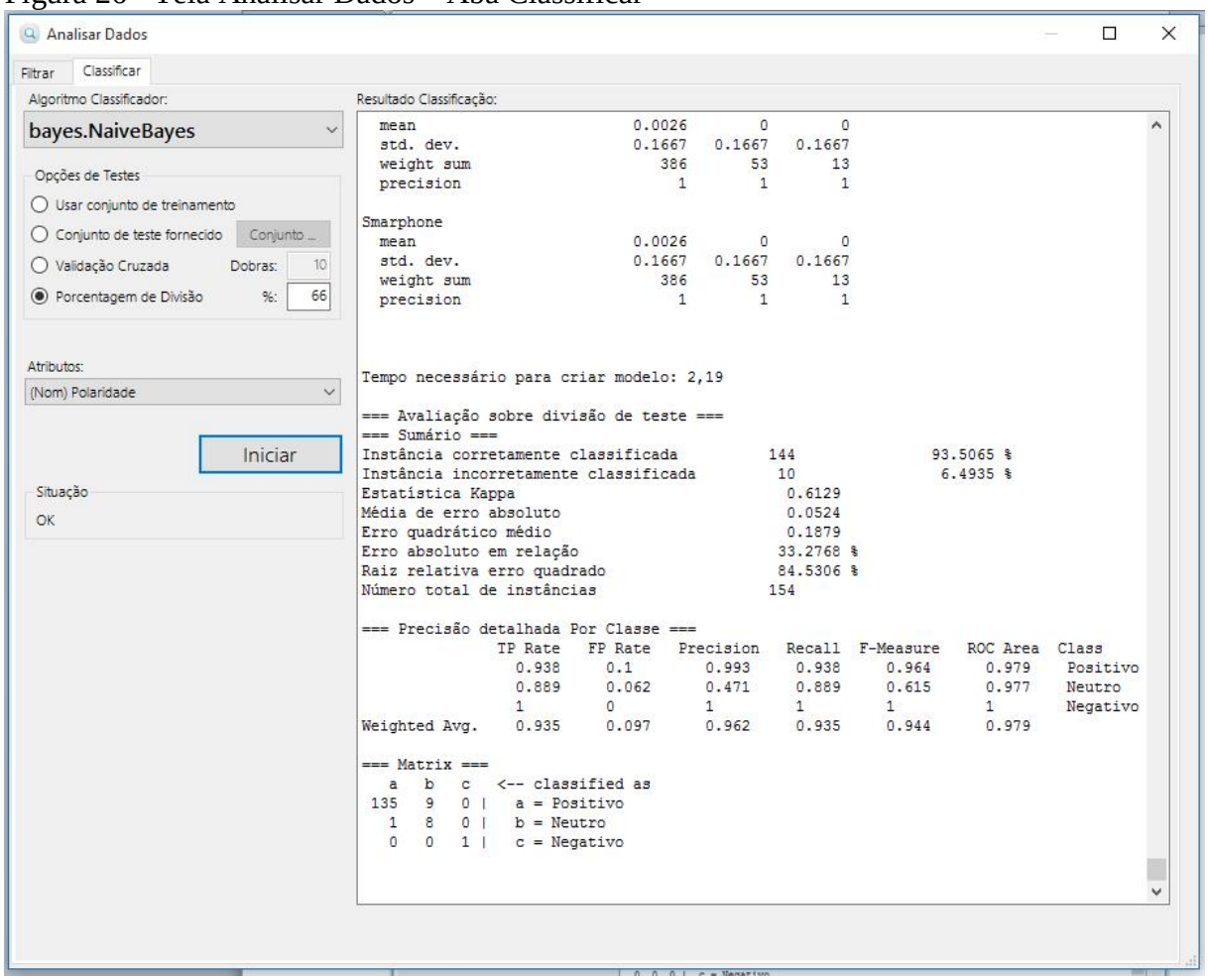
Vale destacar que não há a opção das *stopwords* nessa tela, a opção é usada como na tela comentário, as palavras são buscadas num arquivo posto na raiz da ferramenta, essa busca pelas palavras é sempre realizada quando clicado no botão Aplicar filtro. Não realizando isso seriam pegas palavras vindas por padrão da biblioteca Weka.

Depois desta filtragem os dados dispostos nas instâncias e atributos são reformulados, com isso foi posto o botão Salvar *.arff, para na próxima analise dos mesmos dados o usuário não precise aguardar a filtragem.

O campo % Instância Teste é o percentual de instâncias que o usuário deseja gerar num arquivo arff. Baseado na instância atual o AMD gera um arquivo com todos os atributos atuais e o percentual de instâncias (textos) desejadas, para posterior analise de aprendizagem de máquina de classificação. Para haver compatibilidade de treino e teste os atributos devem ser os mesmos.

Após termos os dados filtrados, o usuário pode seguir para a classificação e análise dos dados, basta clicar na aba Classificar que fica na mesma tela Analisar Dados, ao lado da aba Filtrar, como mostra a Figura 26. Nesta aba ocorre as etapas de extração de padrões e o pós-processamento da Mineração de Dados, comentados na seção 2.2.

Figura 26 - Tela Analisar Dados – Aba Classificar



Fonte: Elaborado pelo autor (2015).

Aba Classificar, seção onde ficam as classificações e análises dos dados filtrados. No lado superior esquerdo o usuário escolhe os algoritmos, comentados na seção 2.3.5, que deseja utilizar para efetuar o treinamento e classificação dos dados, são eles: NaiveBayes do grupo bayes, biblioteca: "weka.classifiers.bayes.NaiveBayes"; SMO do grupo functions, biblioteca: "weka.classifiers.functions.SMO" e NBTree do grupo tree, biblioteca: "weka.classifiers.tree.NBTree". Devido os diversos estudos realizados com o algoritmo C4.5, houve o desejo de utiliza-lo no presente trabalho, no Weka o algoritmo é representado pela biblioteca "weka.classifiers.trees.J48", mas não houve êxito na importação da biblioteca, seguido do mesmo problema com a biblioteca *Evaluation*, conforme capítulo 4.2.

Logo abaixo há 4 opções de testes, na opção "Usar conjunto de treinamento" o AMD usa o conjunto de dados carregado atualmente, lá na aba Filtrar, para criar o modelo de aprendizagem, representando o conhecimento extraído. Na opção "Conjunto de teste fornecido", o usuário define através do botão Conjunto o arquivo com os dados de teste, nesses dados o algoritmo aplicará o seu modelo de aprendizado. Esses testes são rotulados, podendo medir a taxa de acerto do modelo.

A opção "Validação Cruzada", o usuário pode definir o número de dobras, dobras seriam as partes da base de dados. Destas, *X-1* partes são utilizadas para o treinamento e uma como teste. Este processo é repetido *X* vezes, assim cada parte pelos menos uma vez, é um conjunto de teste nas *X* etapas. A correção total é calculada pela média dos resultados em cada etapa, tendo assim uma estimativa de qualidade do modelo de conhecimento gerado (SANTOS et al., 2009).

Na opção "Porcentagem de Divisão", ocorre o processo Split, o usuário define um percentual para base de treinamento e o restante o AMD aloca há uma base de teste. Em seguida ele cria seu modelo de aprendizagem e aplica na base de teste.

Todas as opções são executadas no botão Iniciar, executando:

1. O processo de classificação conforme algoritmo;
2. Executa a avaliação conforme opção de teste selecionada;
3. Os resultados em seguida são mostrados;

Logo abaixo do botão há uma seção mostrando a situação do que o AMD está processando, deixando o usuário informado, quando a situação for "OK" o processo está concluído.

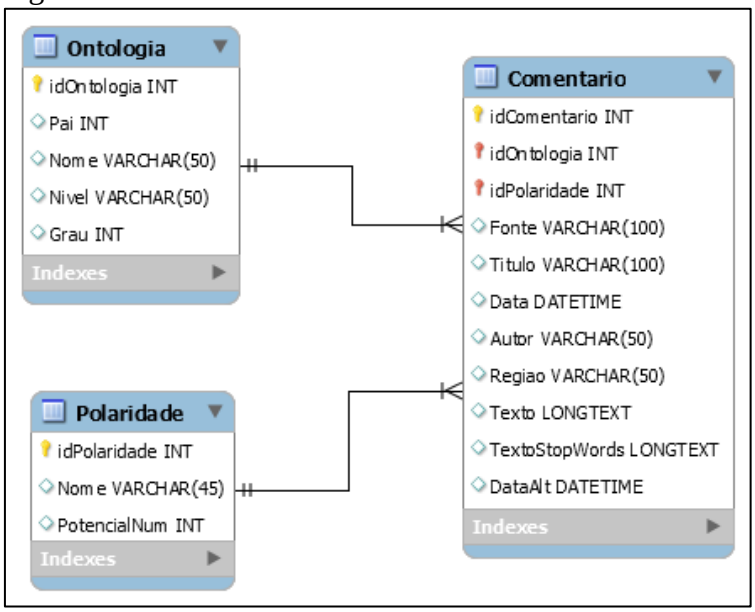
O usuário deve informar a qual classe o algoritmo deverá aplicar seu aprendizado, vale destacar que nos bancos de dados com atributos do tipo *String*, Data esses algoritmos não possuem compatibilidade. Até por isso que é executado o processo de filtragem dos dados com o filtro StringToWordVector, assim, nos dados são atribuídos tipos Nominais e Numéricos, possibilitando o algoritmo atribuir conhecimento/pesos a eles.

No campo Resultado Classificação, constam as informações do processo como, identificação do conjunto de dados, o aprendizado do algoritmo, as avaliações e estatísticas da classificação.

5.2 Modelo da base de dados

Para suportar os recursos do sistema, descritos anteriormente, foi elaborado o diagrama de entidade-relacionamento para o banco de dados seguindo o modelo relacional como mostra a Figura 28. Banco de dados que armazenará as seções de configurações e classificação dos textos.

Figura 27 - Modelo Banco de Dados relacional



Fonte: Elaborado pelo autor (2015).

Na tabela Ontologia (aspectos) serão gravados os registros cadastrados pelo usuário conforme seu tema a ser pesquisado e analisado, toda a estruturada hierárquica é gravado nesta tabela, as definições das colunas são descritas na Tabela 1.

Tabela 1 - Tabela Ontologia

Coluna	Descrição
idOntologia	Código identificador do registro.
Pai	Pai da ontologia cadastrada, esta ontologia pai estará um nível acima na ordenação das ontologias.
Nome	Descrição da Ontologia.
Nivel	Nível da ontologia, conforme ordenação e pai, por exemplo: 001.010.005.
Grau	É a posição na árvore, por exemplo: 1, 2, 3.

Fonte: Elaborado pelo autor (2015).

Na tabela Polaridade (sentimentos) serão gravados os registros cadastrados pelo usuário conforme sua abrangência de sentimentos, por exemplo: Positivo, Neutro e Negativo. Na Tabela 2 constam as definições das colunas.

Tabela 2 - Tabela Polaridade

Coluna	Descrição
idPolaridade	Código identificador do registro.
Nome	Descrição da Polaridade.
PotencialNum	Valor numérico da polaridade, definindo assim a ordenação, por exemplo: 1, -1, 0.

Fonte: Elaborado pelo autor (2015).

A tabela Comentario terá todas as avaliações coletadas pelo usuário, na Tabela 3 destaca as definições das colunas.

Tabela 3 - Tabela Comentario

Coluna	Descrição
idComentario	Código identificador do registro.
idOntologia	Código da Ontologia escolhida pelo usuário na identificação do texto.
idPolaridade	Código da Polaridade escolhido pelo usuário na classificação do texto.
Fonte	Site onde foi retirado o comentário.
Titulo	Titulo que o autor deu ao seu comentário
Data	Data da postagem do comentário
Autor	Autor do comentário.
Regiao	Estado do autor.
Texto	Texto/Avaliação do autor.
TextoStopWords	Texto/Avaliação do autor sem as palavras *stopwords*
DataAlt	Data em que foi feita a coleta.

Fonte: Elaborado pelo autor (2015).

5.3 Testes Realizados

Com a ideia inicial do projeto de coletar os dados do Facebook e Twitter para a obtenção dos dados da web, foi testada a ferramenta desenvolvida no trabalho de FUHR (2014), verificando a disponibilidade de instalação e execução. Inicialmente já houve a notícia

de que a empresa Facebook só disponibilizaria o conteúdo dos *post* da rede social através de pagamento, cortando qualquer possibilidade de coleta *free*.

Mas já no Twitter houve êxito na execução dos *crawlers*, coletados aproximadamente 2.000 *twitts*. Com os dados dentro do banco, depois de estudos no MongoDB, foram extraídos e avaliados, havia conteúdos referentes a produtos de lojas eletrônicas, mas 98% à 99% dos *twitts* seriam propagandas de vendas destes produtos e não de avaliações, tornando esses dados não suficientes para haver avaliações.

Acabou sendo descartada a ferramenta de FUHR (2014) para o objetivo específico do trabalho. Claro que se o assunto foco do trabalho fosse política, filmes entre outros, com certeza a ferramenta seria usada com êxito.

Outra ferramenta testada foi de SCHUSTER FILHO (2013), nesta ferramenta a coleta de dados foi feita diretamente nas lojas eletrônicas mais conhecidas, houve êxito na coleta de informações, mas as informações de avaliação dos produtos não eram acessadas.

Inicialmente não se entendeu o porquê do banco de dados não ter avaliações, mesmo sabendo que os *crawlers* estavam em execução e as configurações de execução estavam corretas. Depois de ter acesso as linhas de código da OntoClipping, foi visto que a ferramenta não tinha acesso ao conteúdo de avaliação, a OntoClipping não foi programada para fazer acesso ao *frame* onde estavam localizadas as avaliações. Com base nessas informações concluiu-se que a OntoClipping não poderia ser usada para o objetivo específico do trabalho.

No desenvolvimento do projeto devido a alguns problemas e testes iniciais optou-se por fazer a coleta de dados manualmente, sem a ajuda de *crawlers* mecânicos, pois testes feitos em textos coletados destas fontes indicaram problemas para classificar. A Figura 29 mostra um comentário coletado manualmente no site de uma loja eletrônica

Figura 28 - Comentário na página da Loja Eletrônica

Fonte: Elaborado pelo autor (2015).

Após a ferramenta concluída, e com alguns dados coletados, a ferramenta começou a ser testada, foram cadastradas 3 polaridades: Positivo, Neutro e Negativo. A tabela Ontologia conta com 30 registros, destes trinta, somente duas (Zenfone 5 – Flex S 1400) têm ligação

direta com os comentários coletados. Como mostra a Tabela 4, foram coletados 251 comentários do Zenfone 5, como mostra Figura 28 um exemplo de comentário do cliente, e 201 comentários do Flex S 1400, totalizando 452 comentários cadastrados manualmente. Em seguida todos os comentários foram também polarizados manualmente.

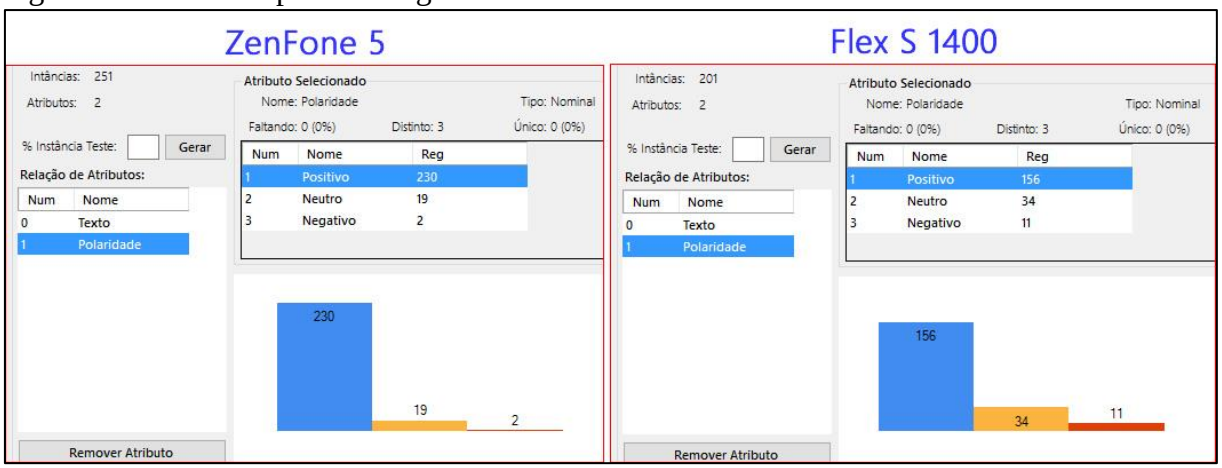
Tabela 4 - Tabela com o total de Instâncias

Polaridade	**Instâncias ZenFone 5**	**Instâncias Flex S 1400**
Positivo	230	156
Neutro	19	34
Negativo	2	11
Total	**251**	**201**

Fonte: Elaborado pelo autor (2015).

Agrupando os comentários por ontologia (aspecto) e polaridade (sentimento), a ontologia Flex S 1400 conta com 156 textos “Positivo”, 34 textos “Neutro” e 11 textos “Negativo”, na ontologia Zenfone 5 conta com 230 textos “Positivo”, 19 textos “Neutro” e 2 textos “Negativo”, como mostra a Figura 30.

Figura 29 - Intâncias por Ontologia



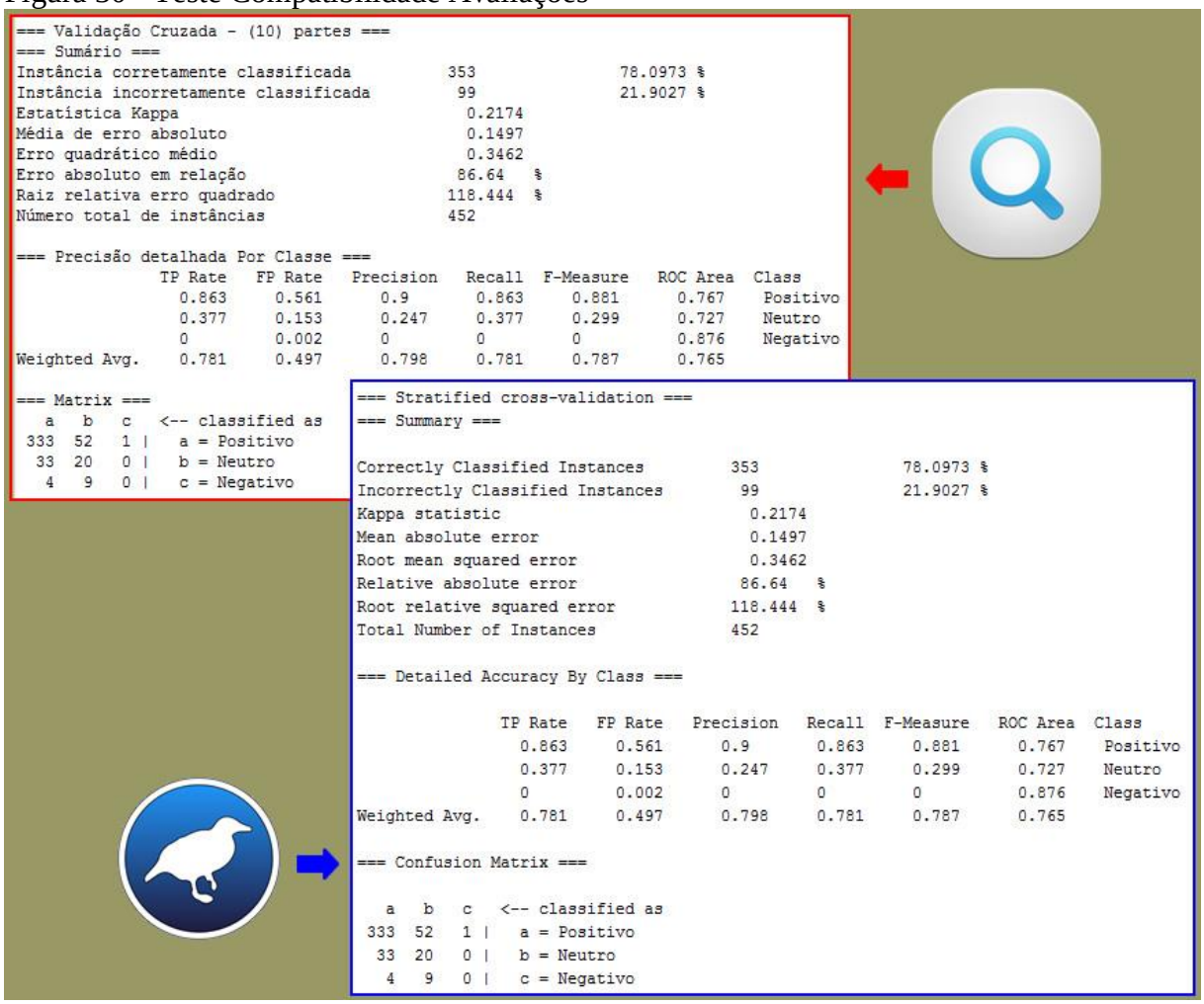
Fonte: Elaborado pelo autor (2015).

Foram testadas as quatros opções de testes desenvolvidas na tela Analisar Dados, ambas tiveram um retorno satisfatório e complementar nas avaliações dos dados, compactando no objetivo do trabalho. Para verificar a veracidade das avaliações foi salva uma instância que já havia sido filtrada pelo StringToWordVector, utilizando os parâmetros padrões do filtro, para um arquivo arff, e assim importado este arquivo na ferramenta Weka e não realizando a filtragem.

Com os dados igualmente carregados foi executada a classificação, a Figura 31 mostra os resultados lado a lado das duas ferramentas, ambas com 452 instâncias e 1.851 atributos, a

opção escolhida de teste foi a Validação Cruzada e o algoritmo NaiveBayes, onde se confirmou a compatibilidade nas avaliações.

Figura 30 - Teste Compatibilidade Avaliações



Fonte: Elaborado pelo autor (2015).

O único diferencial encontrado, que não é mostrado na Figura 31, é o tempo que o algoritmo leva para criar o modelo de aprendizagem de classificação, com isso, outro teste realizado foi verificar a eficiência de criação de modelos dos algoritmos, comparando a AMD com o Weka, os dados importados já haviam sido filtrados, havia um total de 452 instâncias e 1.851 atributos, atributos esses que receberam potencias numéricos nessa classificação, na Tabela 5 a seguir são mostrados os valores encontrados em segundos.

Tabela 5 - Eficiência Criação Modelos dos algoritmos

Algoritmo	**AMD**	**Weka**
bayes.NaiveBayes	1,67 seg.	0,22 seg.
functions.SMO	1,26 seg.	0,14 seg.
trees.NBTree	1.312,61 seg.	95,32 seg.

Fonte: Elaborado pelo autor (2015).

O algoritmo NaiveBayes cria o modelo na AMD em 1,67 seg. e no Weka leva somente 0,22 seg., o algoritmo SMO na AMD cria em 1,26 seg. e no Weka leva somente 0,14 seg., o algoritmo NBTree na AMD cria em 1.312,61 seg. e no Weka leva 95.32 seg. Pode-se ver a grande perda de eficiência em tempo de criação, mas se pensando em números maiores de dados a AMD tende a decepcionar. O algoritmo NBTree é o que mais chama a atenção, tendo um tempo mais significativo convertendo os segundos para minutos, temos aproximadamente 22 min., torna-se uma diferença muito expressiva.

Na Figura 32 é mostrada a avaliação utilizando a opção de teste "Usar conjunto de treinamento", com o algoritmo SMO, não levou um tempo significativo para realizar a avaliação nesta opção de teste. Vale destacar que das 452 instâncias, houve 100% de acerto na classificação das instâncias.

Figura 31 - Conjunto de Treino

```
=== Avaliação no conjunto de treinamento ===
=== Sumário ===
Instância corretamente classificada          452               100      %
Instância incorretamente classificada          0                 0      %
Estatística Kappa                              1
Média de erro absoluto                         0.2222
Erro quadrático médio                          0.2722
Erro absoluto em relação                     128.7773 %
Raiz relativa erro quadrado                   93.1412 %
Número total de instâncias                   452

=== Precisão detalhada Por Classe ===
               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1         1         1          1        Positivo
                 1         0          1         1         1          1        Neutro
                 1         0          1         1         1          1        Negativo
Weighted Avg.    1         0          1         1         1          1

=== Matrix ===
   a   b   c   <-- classified as
 386   0   0 |   a = Positivo
   0  53   0 |   b = Neutro
   0   0  13 |   c = Negativo
```

Fonte: Elaborado pelo autor (2015).

Na Figura 33 mostra o resultado da opção de teste "Conjunto de teste fornecido", o arquivo (conjunto) origem é da mesma base anteriormente testada, foram pegos 30% das instâncias e gerado um arquivo teste, também não houve um tempo significativo de espera. Das 135 instâncias testadas, houve um índice de 92% de instâncias corretamente classificadas, valor muito considerável.

Figura 32 - Conjunto de Teste

```
=== Avaliação no conjunto de teste ===
=== Sumário ===
Instância corretamente classificada          125               92.5926 %
Instância incorretamente classificada         10                7.4074 %
Estatística Kappa                             -0.0128
Média de erro absoluto                         0.2387
Erro quadrático médio                          0.3009
Erro absoluto em relação                     231.1749 %
Raiz relativa erro quadrado                  237.3561 %
Número total de instâncias                   135

=== Precisão detalhada Por Classe ===
               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.933     1          0.992     0.933     0.962      0.466    Positivo
                 0         0.06       0         0         0          0.474    Neutro
                 0         0.007      0         0         0          ?        Negativo
Weighted Avg.    0.926     0.993      0.985     0.926     0.954      0.466

=== Matrix ===
   a   b   c   <-- classified as
 125   8   1 |   a = Positivo
   1   0   0 |   b = Neutro
   0   0   0 |   c = Negativo
```

Fonte: Elaborado pelo autor (2015).

A Figura 34 mostra a "Validação Cruzada – 10 Dobras" sendo a opção de teste, neste teste ocorre a maior espera por resultados, espera devido os testes ocorrerem conforme o número de dobras, como nesta figura, ocorreram 10 testes sendo avaliados, baseado nesses resultados é feita uma média, esse resultado é mostrado.

Figura 33 - Validação Cruzada

```
=== Validação Cruzada - (10) partes ===
=== Sumário ===
Instância corretamente classificada          372               82.3009 %
Instância incorretamente classificada         80               17.6991 %
Estatística Kappa                              0.0605
Média de erro absoluto                         0.2689
Erro quadrático médio                          0.3475
Erro absoluto em relação                     155.6447 %
Raiz relativa erro quadrado                  118.916  %
Número total de instâncias                   452

=== Precisão detalhada Por Classe ===
               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.951     0.894      0.862     0.951     0.904      0.528    Positivo
                 0.094     0.053      0.192     0.094     0.127      0.5      Neutro
                 0         0          0         0         0          0.569    Negativo
Weighted Avg.    0.823     0.77       0.758     0.823     0.787      0.526

=== Matrix ===
   a   b   c   <-- classified as
 367  19   0 |   a = Positivo
  48   5   0 |   b = Neutro
  11   2   0 |   c = Negativo
```

Fonte: Elaborado pelo autor (2015).

A opção de teste “Porcentagem de divisão – 66%” é mostrada na Figura 35.

Figura 34 - Divisão de Teste

```
=== Avaliação sobre divisão de teste ===
=== Sumário ===
Instância corretamente classificada          154               100      %
Instância incorretamente classificada          0                 0      %
Estatística Kappa                              1
Média de erro absoluto                         0.2222
Erro quadrático médio                          0.2722
Erro absoluto em relação                     141.1854 %
Raiz relativa erro quadrado                  122.4712 %
Número total de instâncias                   154

=== Precisão detalhada Por Classe ===
               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1         1         1          1        Positivo
                 1         0          1         1         1          1        Neutro
                 1         0          1         1         1          1        Negativo
Weighted Avg.    1         0          1         1         1          1

=== Matrix ===
   a   b   c   <-- classified as
 144   0   0 |   a = Positivo
   0   9   0 |   b = Neutro
   0   0   1 |   c = Negativo
```

Fonte: Elaborado pelo autor (2015).

Do total do conjunto de treinamento, 66% dos dados ficam treinos e o resto 34% fica como teste, assim os dados de treinamento será aplicado nestes 34%, o tempo de avaliação se mostrou insignificante para a demanda. Vale destacar que nas 154 instâncias testes avaliadas, houve 100% de instâncias corretamente classificadas.

6 CONCLUSÃO

Este trabalho de conclusão de curso, com base nas diversas fontes bibliográficas, documentou as etapas da mineração de dados como a descoberta de conhecimento, o pré-processamento dos dados onde são realizadas limpezas de ruídos e atribuídos potencias numéricos as palavras, a extração de padrões com a aplicação de algoritmos e o pós-processamento, etapa que é feita a avaliação dos dados classificados.

Foram também destacadas as diversas ferramentas de mineração de dados existentes no mercado, mas como visto, elas possuem complexidade de entendimento e linguagem inacessível para muitos. Isso mostra que o desenvolvimento realizado e documentado nas seções anteriores faz de grande valia para futuros usuários interessados em descobrir conhecimento nos textos.

A conclusão da ferramenta mostra que o objetivo principal deste trabalho foi alcançado. A definição dos aspectos nos textos é realizada através do cadastro de ontologia e o sentimento através do cadastro de polaridade. A utilização de recursos da ferramenta Weka para realizar o processo de mineração de dados foi alcançada através de documentações disponibilizadas na *web*.

Os recursos disponibilizados visam facilitar o processo de análise de textos, a definição de aspectos e alvos do sentimento contido nos textos, além de automatizar a conversão dos dados no formato necessário para aplicação dos algoritmos. Boa parte dos *softwares* para mineração de dados exige que o usuário realize estas tarefas de forma manual e que o mesmo tenha conhecimento sobre estruturas de arquivos e algoritmos de aprendizagem de máquina.

Para exportar os dados para o Weka, o AMD disponibiliza o módulo de exportação arquivo arff, sendo possível utilizar de outros recursos do Weka, recursos esses que a AMD não possui. É possível filtrar os textos por aspecto e classes num conjunto de dados polarizados, na sequência visualizando as avaliações e resultados das classificações. A ferramenta mostra um ponto forte no processo da captura dos dados, caso os dados estejam cadastrados dentro da ferramenta, eles podem ser carregados para classificação sem a geração de arquivo arff, onde na Weka é obrigatório já que não possui base vinculada.

Uma das limitações percebidas diz respeito ao tempo de processamento, que se mostrou superior a outros softwares, principalmente ao Weka, no qual foi comparado. Estes tempos podem ser reduzidos pela aplicação de técnicas de programação paralela ou a partir de melhorias na arquitetura da ferramenta desenvolvida.

Referente ao treinamento realizado pelos algoritmos no conjunto de dados houve a tentativa de salvar esse treinamento para futuras avaliações, para não haver a necessidade de esperar a classificação que tem o tempo maior de espera. Estudando o recurso, chegou-se à conclusão que ao salvar o treinamento devem ser exportados todos os atributos da avaliação atual com seus potenciais numéricos, e nas futuras avaliações os atributos presentes devem ser os mesmos do treinamento exportado, a necessidade de definir outros algoritmos de filtro, não somente o StringToWordVector. Verificando a complexidade desta modalidade essa ideia foi desconsiderada do trabalho.

Acreditando que a ferramenta desenvolvida cumpre com todas as etapas da mineração de dados, e esta proposta pode ser aperfeiçoada em trabalhos futuros, podem ser acrescentados mais algoritmos de pré-processamento e pós-processamento e maior manipulação dos dados. Facilitar ainda mais os processos com opções de configurações salvas em cada etapa do processo de mineração, salvar resultados para visualização de gráficos estatísticos e gerar dados de erros.

Um aspecto considerado interessante é a importação dos dados para a ferramenta AMD, hoje ela está projetada para a coleta manual, mas definir configurações de importação a arquivos ou banco de dados à torna mais robusta.

Até o presente momento o sistema realiza parte das funcionalidades esperadas de uma ferramenta para classificação automática. São testados os algoritmos de aprendizado de máquina e é possível definir aspectos associados a textos. Entretanto, em trabalhos futuros,

espera-se manter de forma permanente os resultados do aprendizado e aplicar os algoritmos a novas publicações.

7 REFERÊNCIAS

AMORIM, Thiago. **Conceitos, técnicas, ferramentas e aplicações de Mineração de Dados para gerar conhecimento a partir de bases de dados**. Monografia (Bacharel em Ciência da Computação). Universidade Federal de Pernambuco, 2006.

ARAÚJO, Matheus; GONÇALVES, Pollyanna; BENEVENUTO, Fabrício. **Métodos para Análise de Sentimentos no Twitter.** 2013.

ARRIAL, Roberto Ternes. **Predição de RNAs não-codificadores no transcriptoma do fungo Paracoccidioides brasiliensis usando aprendizagem de máquina**. 2008. Tese de Doutorado. Instituto de Biologia.

BASGALUPP, Márcio Porto. **LEGAL-Tree: um algoritmo genético multi-objetivo para indução de árvores de decisão.** Diss. Universidade de São Paulo, 2010.

BECKER, Karin. Introdução à Mineração de Opiniões. **XXXIV Congresso da Sociedade Brasileira de Computação.** Porto Alegre, RS, cap. 4, pág. 125-176, 2014.

BECKER, Karin; TUMITAN, Diego. Introdução à Mineração de Opiniões: Conceitos, Aplicações e Desafios. **Simpósio Brasileiro de Banco de Dados**, 2013.

BENEVENUTO, Fabrício; ALMEIDA, Virgílio; HORIZONTE-BRASIL, Belo. **Uma análise empírica de interações em redes sociais**. Proceedings of the XXIV Concurso de teses e dissertações (CTD), Natal, Brazil, 2011.

BERNARDO, André. Como os algoritmos dominaram o mundo. **Galileu**, São Paulo, 20 Maio, 2014. Disponível em: < http://revistagalileu.globo.com/Revista/noticia/2014/05/como-os-algoritmos-dominaram-o-mundo.html>. Acesso em: 15 mai. 2015.

BRITO, Edivaldo. Twitter recupera e cola no Instagram em total de usuários: 302 milhões. **TechTudo**, São Paulo, 29 Abril, 2015. Disponível em: < http://www.techtudo.com.br/noticias/noticia/2015/04/twitter-recupera-e-cola-no-instagram-em-total-de-usuarios-302-milhoes.html>. Acesso em: 15 mai. 2015.

BULSING, Gabriel Merten. **Ferramenta para extração de dados semi-estruturados para carga de um Big Data**. UNISC, Santa Cruz do Sul, RS, 2013.

CAMILO, Cássio Oliveira; SILVA, João C. **Mineração de dados: Conceitos, tarefas, métodos e ferramentas.** Universidade Federal de Goiás (UFC), p. 1-29, 2009.

CARVALHO, Cláudio Omar Correa. **Robô de Captura e Indexação de Textos para Clipagem Online com base em Ontologias**. UNISC, Santa Cruz do Sul, RS, 2012.

CARVALHO, Jonnathan dos Santos. **Uma estratégia estatística e evolutiva para Mineração de Opiniões em *Tweets***. 2014.

CHAPMAN, Pete et al. **CRISP-DM 1.0 Step-by-step data mining guide**. 2000.

CIN. Weka.doc. **Centro de Informática de UFPE - CIn**, Pernambuco, 23 Abril, 2004. Disponível em: < www.cin.ufpe.br/~mcps/IA/IA2004.1/weka.doc>. Acesso em: 03 jun. 2015.

DA SILVA, Wilson Carlos; MARTINS, Luiz Eduardo Galvão. **PARADIGMA: Uma Ferramenta de Apoio à Elicitação e Modelagem de Requisitos Baseada em Processamento de Linguagem Natural**. WER, v. 8, p. 140-151, 2008.

DIAS, Maria Madalena. Parâmetros na escolha de técnicas e ferramentas de mineração de dados. **Acta Scientiarum. Technology**, v. 24, p. 1715-1725, 2002.

DIAS, Maria Abadia Lacerda; DE GOMENSORO MALHEIROS, Marcelo. **Extração Automática de Palavras-chave de Textos da Língua Portuguesa.** Centro Universitário UNIVATES, 2005.

FAYYAD, Usama; PIATETSKY-SHAPIRO, Gregory; SMYTH, Padhraic. From data mining to knowledge discovery in databases. **AI magazine**, v. 17, n. 3, p. 37, 1996.

FREITAS, Cláudia. A sentiment lexicon for portuguese natural language processing. **Revista Brasileira de Linguística Aplicada**, v. 13, n. 4, p. 1031-1059, 2013.

FUHR, Bruno Edgar. **Desenvolvimento de uma ferramenta de coleta e armazenamento de dados para Big Data.** UNIVATES, Lajeado, RS, 2014.

KHANALE, P. B. et al. **WEKA: A Dynamic Software Suit for Machine Learning & Exploratory Data Analysis**. BIOINFO, Volume 1, Issue 1, pág.01-05, 2011.

IKVM. **IKVM.NET - User's_Guide**. Disponível em: <http://sourceforge.net/p/ikvm/wiki/User's_Guide/>. Acesso em: 23 out. 2015.

LAKATOS, Eva Maria; MARCONI, Marina de Andrade. **Fundamentos de metodologia científica**. 7. ed. São Paulo: Atlas, 2010.

LINGUATECA.PT. **Listas de palavras frequentes em português, *Lists of Portuguese stopwords***. Disponível em: < http://www.linguateca.pt/chave/stopwords/folha.MF300.txt >. Acesso em: 07 out. 2015.

MONGODB, Inc. **Introduction to MongoDB**. Disponível em: <https://docs.mongodb.org/manual/core/introduction/>. Acesso em: 02 nov. 2015.

MONO. **Mono Project**. Disponível em: < http://www.mono-project.com/>. Acesso em: 23 out. 2015.

REZENDE, Solange Oliveira. Mineração de Dados. **XXV Congresso da Sociedade Brasileira de Computação**, UNISINOS, São Leopoldo, RS, pág. 397-433, 2005.

REZENDE, S. O. et al. **Mineração de Dados, in REZENDE,** S. O. (Eds.), Sistemas Inteligentes, Editora Manole Ltda., p.307-335. 2003.

REZENDE, Solange O.; MARCACINI, Ricardo M.; MOURA, Maria F. O uso da mineração de textos para extração e organização não supervisionada de conhecimento. **Revista de Sistemas de Informação da FSMA**, v. 7, p. 7-21, 2011.

RODRIGUES, Daniel Henriques. **Construção Automática de um Dicionário Emocional para o Português**. 2009.

SANTOS, Aline Graciela Lermen dos; BECKER, Karin; MOREIRA, Viviane. **Mineração de emoções em textos multilíngues usando um corpus paralelo.** SBBD Proceedings, Curitiba, PR, pág. 79, 2014.

SANTOS, Leandro Matioli et al. Twitter, análise de sentimento e desenvolvimento de produtos: Quanto os usuários estão expressando suas opiniões?. **Revista PRISMA.COM**, n. 13, 2011.

SANTOS, Luciano Drosda M. dos, et al. **Procedimentos de Validação Cruzada em Mineração de Dados para ambiente de Computação Paralela.** ERAD 2009, Caxias do Sul, pág. 233-236, 2009.

SÁPIRAS, Leonardo Augusto; BECKER, Karin. **Identificação de aspectos de candidatos eleitorais em comentários de notícias.** [201-]

SILVA, Nelson Rocha; LIMA, Diego; BARROS, Flávia. SAPair: Um Processo de Análise de Sentimento no Nível de Característica. In: **4nd International Workshop on Web and Text Intelligence (WTI'12)**, Curitiba. 2012.

SCHMIIT, Vinícius Fernandes. **Uma Análise Comparativa de Técnicas de Aprendizagem de Máquina para prever a Popularidade de Postagens no Facebook**. UNISC, Santa Cruz do Sul, RS, 2013.

SCHUSTER FILHO, Roberto Antonio. **Adaptação Temporal e Qualitativa sobre Mecanismos de Clipagem Eletrônica.** UNISC, Santa Cruz do Sul, RS, 2013.

WAIKATO. Weka Knowledge Explorer. **The University of WAIKATO.** Disponível em: < http://www.cs.waikato.ac.nz/ml/weka/>. Acesso em: 03 jun. 2015.